Karma Provenance Retrieval and Visualization Plugin
For Cytoscape
User Manual v1.2.0
Feb 24th, 2012

Copyright 2012 The Trustees of Indiana University

This document contains instructions for using the Karma provenance retrieval and visualization plugin version 1.2.0, which provides core capability to retrieve provenance information from a Karma provenance system and visualize the returned graph. Karma provenance retrieval and visualization plugin is licensed under Apache License, Version 2.0 (the "License") (http://www.apache.org/licenses/LICENSE-2.0). The code is copyrighted and copyright owned by The Trustees of Indiana University. Karma provenance retrieval and visualization plugin is a product of the Data to Insight Center at Indiana University. See http://pti.iu.edu/d2i/provenance for more information.

Contents

1. Introduction 4
2. Software Dependencies 4
   2.1 Service dependencies 4
   2.2 Installation dependencies 5
3. Installing Plugin 5
   3.1 Installing under Mac/Linux/Unix OS 5
   3.2 Installing under Windows OS 6
4. Configuring plugin properties 7
   4.1 Updating the configuration files to connect Karma provenance retrieval and visualization plugins to Karma service 7
   4.2 Updating the configuration file for Karma visualization plugin 8
5. Using the Karma provenance retrieval plugin 8
6. Using the Karma visualization plugin 10
   6.1 Load the Karma provenance XML file 10
   6.2 Applying layout algorithms to the Karma provenance graph 13
      6.2.1 X axis sort 14
      6.2.2 Network Simulator Layout 15
      6.2.3 Network Simulator Layout Extension 1 16
   6.3 Navigating the Karma provenance graph 21
   6.4 Playing movie 22
   6.5 Saving graph as XML 23
   6.6 Getting data provenance history 24
   6.7 Creating visual style 25
   6.8 Creating an abstract view 26
      6.8.1 Clustering neighbor nodes 26
      6.8.2 Navigate between graph and sub-graphs 29
      6.8.3 Compress Process/Artifact 30

1. Introduction

We have developed two plugins to Cytoscape to visualize and navigate provenance information contained in Karma provenance system.

The Karma provenance retrieval plugin is used to retrieve provenance graphs from a remote Karma server to your local machine. In Cytoscape, this plugin is executed by clicking on the blue "Karma" icon in the Cytoscape toolbar after the plugin is installed. The second plug-in is the Karma visualization plugin, which allows the user to visualize and manipulate graphs downloaded using the retrieval plugin. The Karma visualization plugin is displayed as an orange "Geni" icon in the Cytoscape toolbar.

Visualization of provenance data is useful for manipulating very large provenance graphs, for displaying different views, and for interactivity. This can help a user to navigate their experiment information with a mental map of what is going on in the experiment, to compare different experiment runs quantitatively, and to do model selection with an effective collaboration between the user and the discovery system.
Karma is a standalone system that can be added to existing cyberinfrastructure for purposes of collection and representation of provenance data. The Karma query plugin aims to provide a GUI component that queries provenance information of scientific experiments from the Karma provenance repository. The Karma server is accessible via either a webservice API or RabbitMQ enterprise bus, and our provenance retrieval plugin supports both access methods.

Cytoscape (http://www.cytoscape.org/) is an open source software platform for complex network analysis and visualization. We use Cytoscape because of its support for detail and overlaying visualizations with additional annotations. We developed our visualization tool as a plugin that can generate the provenance graph visualization directly from the Karma provenance information using an XML, and provide control of the navigation process.

2. Software Dependencies

2.1 Service dependencies

The Karma plugin retrieves provenance as graphs from the Karma provenance server. To use the Cytoscape plugins, one will need to either setup a Karma server or process a log file using an established Karma service such Data to Insight Center's NetKarma service on the GRNOC server. To set up a Karma server, please refer to the Karma Provenance System user guide at
http://pti.iu.edu/d2i/provenance_karma

If an instance of Karma service (either hosted as a web service or as a standalone service using the RabbitMQ messaging bus) already exists, contact your system administrator for obtaining access.

2.2 Installation dependencies

The Karma provenance retrieval and visualization plugin v1.1.0 has been tested with the following software packages on which it has a dependency. These packages will need to be installed before using the visualization plugin:

1) Java Development Kit (JDK) v5 or v6
   http://java.sun.com
2) Cytoscape v2.8.2
   http://www.cytoscape.org/

3. Installing Plugin

==========

Download the plugin package as a zip file from:
http://pti.iu.edu/d2i/provenance_karma

3.1 Installing under Mac/Linux/Unix OS

1) Check out the plugin package from our SVN repository:

```
svn co https://karmatool.svn.sourceforge.net/svnroot/karmatool/karma/trunk/visualization visualization
```

2) Building from source code (optional)

This step can be skipped, since the plugin package already includes pre-build plugin jar files. If one wants to build from source code, the ant properties in the file "build.properties" needs to be set.

To build the Karma visualization plugin, the file "visualization/OPM_visualization/build.properties" needs to be edited:

```
<!-- Define the Cytoscape directories -->
cytoscape.dir= Cytoscape_v2.8.2      //the directory where  Cytoscape is installed on your computer
```

In the directory of OPM_visualization, type in the command

```
ant
```

If this succeeds, a jar file named "KarmaGraph.jar" should be generated.

To build the Karma retrieval plugin, the file "visualization/Karma_query/build.properties" needs to be edited:

```
<!-- Define the Cytoscape directories -->
cytoscape.dir=  Cytoscape_v2.8.2      //the directory where  Cytoscape is installed on your computer
```

In the directory of Karma_query, type in the command

```
ant
```

If this succeeds, a jar file named "KarmaRetrieval.jar" should be generated.

3) Copy the jar files into the plugins directory under Cytoscape_v2.8.2

Deploy the Karma visualization plugin:

```
cp visualization/OPM_visualization/KarmaGraph.jar  Cytoscape_v2.8.2/plugins/
```

Deploy the dependent libraries for Karma visualization plugin:

```
cp visualization/OPM_visualization/lib/*  Cytoscape_v2.8.2/plugins/
```

Deploy the Karma provenance retrieval plugin:

```
cp visualization/Karma_query/KarmaRetrieval.jar  Cytoscape_v2.8.2/plugins/
```

Deploy the dependent libraries for Karma provenance retrieval plugin:

```
cp visualization/ Karma_query /lib/*  Cytoscape_v2.8.2/plugins/
```

Note: some libraries are shared by Karma visualization plugin.

4) Copy the configuration files into the plugins directory under Cytoscape_v2.8.2

Create a new directory named "config" under cytoscape's "plugins" directory:

```
mkdir Cytoscape_v2.8.2/plugins/config
```

Deploy the configuration file for the Karma visualization plugin:

```
cp visualization/OPM_visualization/config/*  Cytoscape_v2.8.2/plugins/config
```

Deploy the configuration file for the Karma provenance retrieval plugin:

```
cp visualization/Karma_query/config/*  Cytoscape_v2.8.2/plugins/config
```

5) After installing the two plugins, the "plugins" directory under Cytoscape_v2.8.2 should look like:

```
- Cytoscape_v2.8.2 HOME DIRECTORY
---plugins/
--------KarmaGraph.jar
--------KarmaRetrieval.jar
--------Other jar libraries
--------config/
------------karmaQueryConfig.txt
------------karmaVisConfig.txt
------------pluginConfig.xml
```

3.2 Installing under Windows OS

1) Checkout the plugin package from our SVN repository:

```
svn co https://karmatool.svn.sourceforge.net/svnroot/karmatool/karma/trunk/visualization visualization
```

2) Building from source code (optional)

Same as step 2 in Section 3.1.

3) Copy the jar files into the plugins directory under Cytoscape_v2.8.2

Install Karma visualization plugin:

```
copy visualization/OPM_visualization/KarmaGraph.jar  "C:\Program files\ Cytoscape_v2.8.2\plugins\"
```

Deploy libraries for Karma visualization plugin:

```
copy visualization/OPM_visualization/lib/*  "C:\Program files\ Cytoscape_v2.8.2\plugins\"
```

Deploy Karma provenance retrieval plugin:

```
copy visualization/Karma_query/KarmaRetrieval.jar  "C:\Program files\ Cytoscape_v2.8.2\plugins\"
```

Deploy libraries for Karma provenance retrieval plugin:

```
copy visualization/ Karma_query /lib/*  "C:\Program files\ Cytoscape_v2.8.2\plugins\"
```

4) Copy configuration files into the plugins directory under Cytoscape_v2.8.2

Create a new directory named "config" under cytoscape's "plugins" directory:

```
mkdir "C:\Program files\ Cytoscape_v2.8.2\plugins\config"
```

Deploy the configuration file for Karma visualization plugin:

```
copy visualization/OPM_visualization/config/*  "C:\Program files\ Cytoscape_v2.8.2\plugins\config"
```

Deploy the configuration file for Karma provenance retrieval plugin:

```
copy visualization/Karma_query/config/*  "C:\Program files\ Cytoscape_v2.8.2\plugins\config"
```

5) After installing the two plugins, the "plugins" directory under Cytoscape_v2.8.2 will look like:

```
-Cytoscape_v2.8.2 HOME DIRECTORY
---plugins/
--------KarmaGraph.jar
--------KarmaRetrieval.jar
--------Other jar libraries
--------config/
------------karmaQueryConfig.txt
------------karmaVisConfig.txt
------------pluginConfig.xml
```

4. Configuring plugin properties

==========

4.1 Updating the configuration files to connect Karma provenance retrieval and visualization plugins to Karma service

The following describes how to configure the provenance retrieval and visualization plugins and connect them to a running Karma service.

There are two options (Axis2 webservice or RabbitMQ messaging system) to connect to a Karma server, and all of the configuration information is stored in the file "karmaQueryConfig.txt" for Karma Retrieval plugin and in the file "karmaQueryConfig.txt" for Karma visualization plugin. These configuration files will be loaded at the startup of either plugin, and the configuration settings will be displayed in the configuration panel. The default configuration settings can be modified and will be saved to the configuration files automatically.

Besides making configuration changes through the configuration panel in Cytoscape, the contents in the configuration file can be modified. The file "karmaQueryConfig.txt" has identical configuration entries as the file "karmaVisConfig.txt".
Set up the configuration for connecting via the Axis2 webservice:
There is only one property that needs to be set when using the Axis2 server:

axis2.serviceURL – enter the URL to the Karma v3.2.1 webservice

Set up the configuration for connecting via RabbitMQ messaging system:
There are several properties that must be set to connect using the RabbtiMQ server:

messaging.username – username of RabbitMQ
messaging.password – password of RabbitMQ
messaging.hostname – hostname or IP address
messaging.hostport – port

4.2 Updating the configuration file for Karma visualization plugin

The "pluginConfig.xml" has properties that control the labeling of nodes. Karma visualization plugin automatically import all the annotations into Cytoscape graph as attributes. User can choose the label of nodes from one of the imported attributes by configuring the following properties:

visualAttributes – configuration for the properties related to the rendering of the graph, includes:
nodeLabel – for telling which attribute of the nodes to show as labels. The configuration for PROCESS and ARTIFACT are separated:
processLabel – the label of process nodes
artifactLabel – the label of artifact nodes
Each group of configuration has a list of attribute names listed with a priority from the highest to the lowest:
attributeName – which attribute of the nodes to show as labels

For example, if the attribute "objectValue" or the attribute "ID" is to be used as the label of artifacts, the following configuration can be setup, where the "ID" attribute will be used when the "objectValue" attribute is not present.

```
<artifactLabel>
        <attributeName>objectValue</attributeName>
        <attributeName>ID</attributeName>
</artifactLabel>
```

5. Using the Karma provenance retrieval plugin

==========

1) Run Cytoscape and click on the Karma icon in the toolbar.

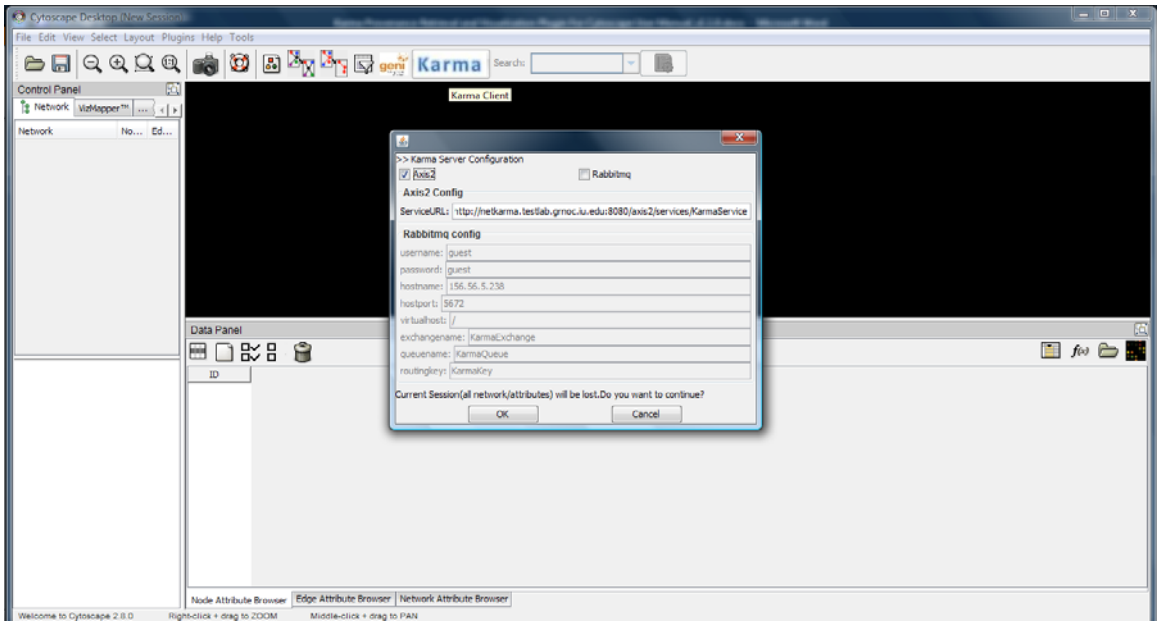

Figure 5.1 Click on Karma toolbar

Choose the connection method and configure its connection parameters (Choosing “Axis2”, the “ServiceURL” in the “Axis2 Config” panel needs to be configured; choosing “Rabbitmq”, all parameters in the “Rabbitmq config” panel need to be configured).

2) Next, click the “OK” button and a new dialog window will appear which prompts for the workflowID.

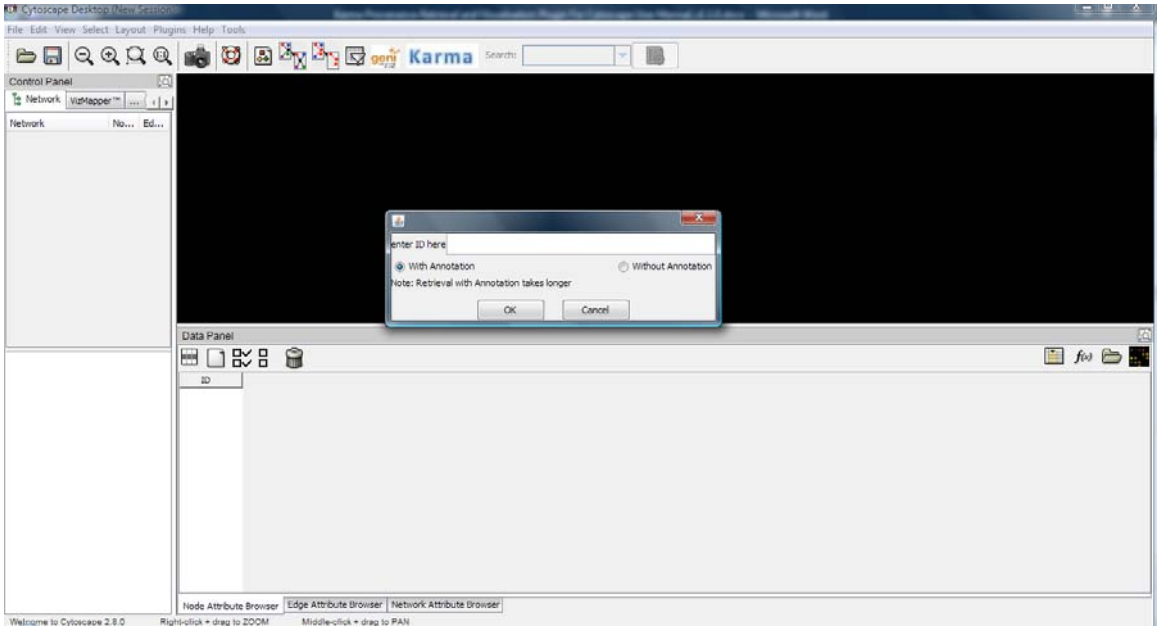

Figure 5.2 Enter the workflow ID

Enter the workflowID of the intended graph, and select whether annotations are to be included in the graph or not, then press the OK button to continue.

3) After the plugin has retrieved the Karma provenance XML, there will be a prompt to choose the name and location to save the XMLfile.

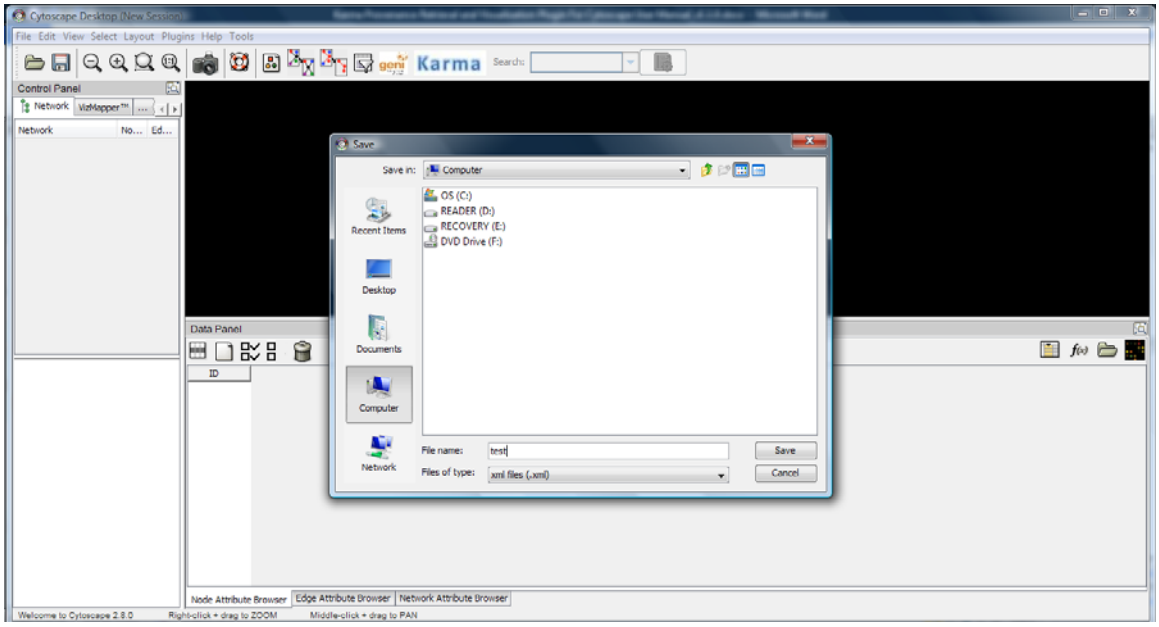

Figure 5.3 Save the OPM XML file

Note 1: The time required to retrieve the graph varies depending on the size and complexity of the XML file. Retrieving a graph without annotations can significantly reduce the querying time. For large provenance graphs we recommend retrieving without annotations since the Karma visualization plugin will load the annotations “On Demand” while navigating the provenance graph.

Note 2: While retrieving the XML file, Cytoscape will appear to be blocked, but will continue to work right after the file is retrieved successfully. Please be patient while waiting for a graph to be downloaded.

6. Using the Karma visualization plugin

==========

6.1 Load the Karma provenance XML file

1) Click on the GENI toolbar

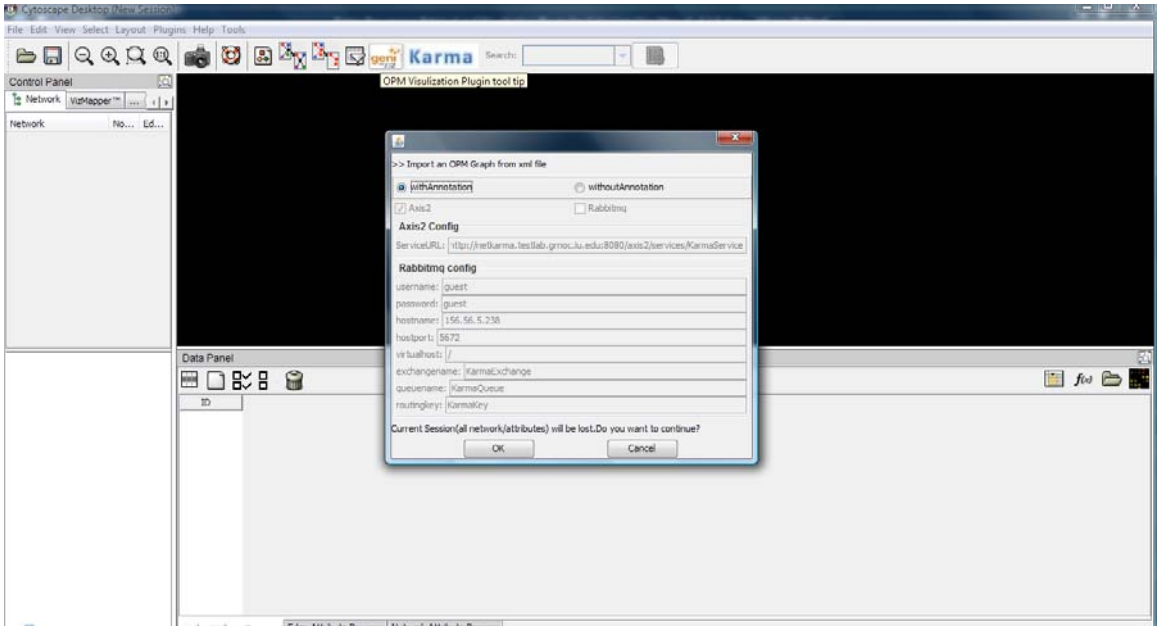

Figure 6.1.1 Click on the GENI toolbar

Select whether to import the provenance xml with annotations or without annotations. When importing provenance without annotations, the Karma visualization plugin will set up a background connection to the Karma server, and load both annotations and the registry level information “On Demand” (when any node is selected from the provenance graph during navigation, the Karma visualization will retrieve more information for it).

2) If the provenance graph is imported without annotations, the background connection method needs to be chosen and the connection parameters need to be configured.

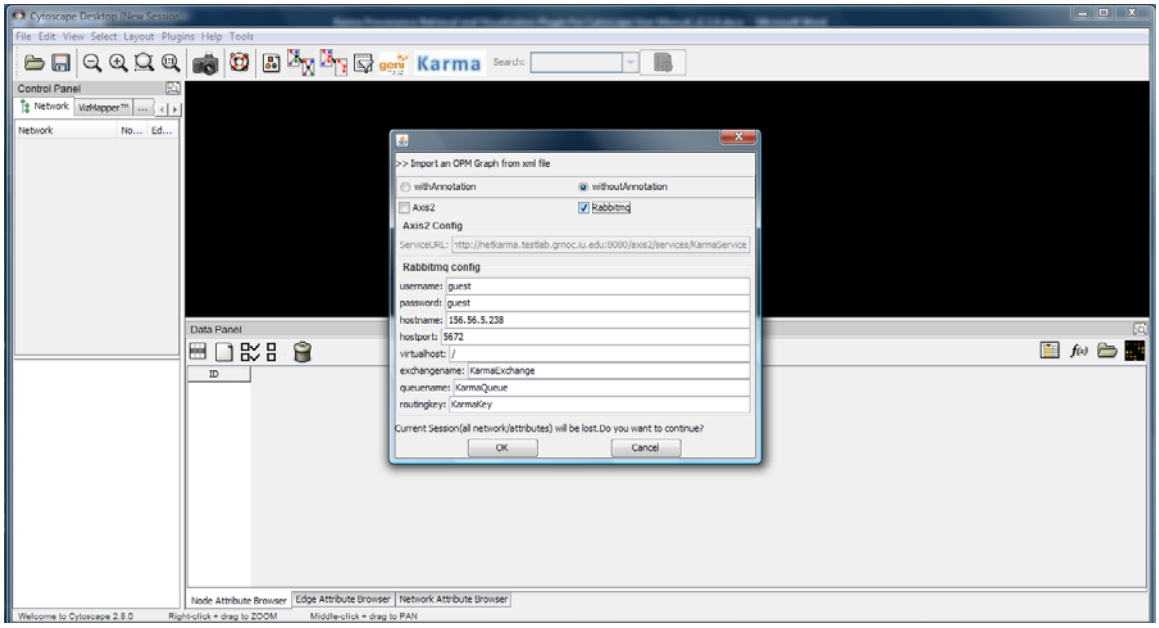

Figure 6.1.2 Configure the connection to Karma Server

3) Select the Karma provenance XML file

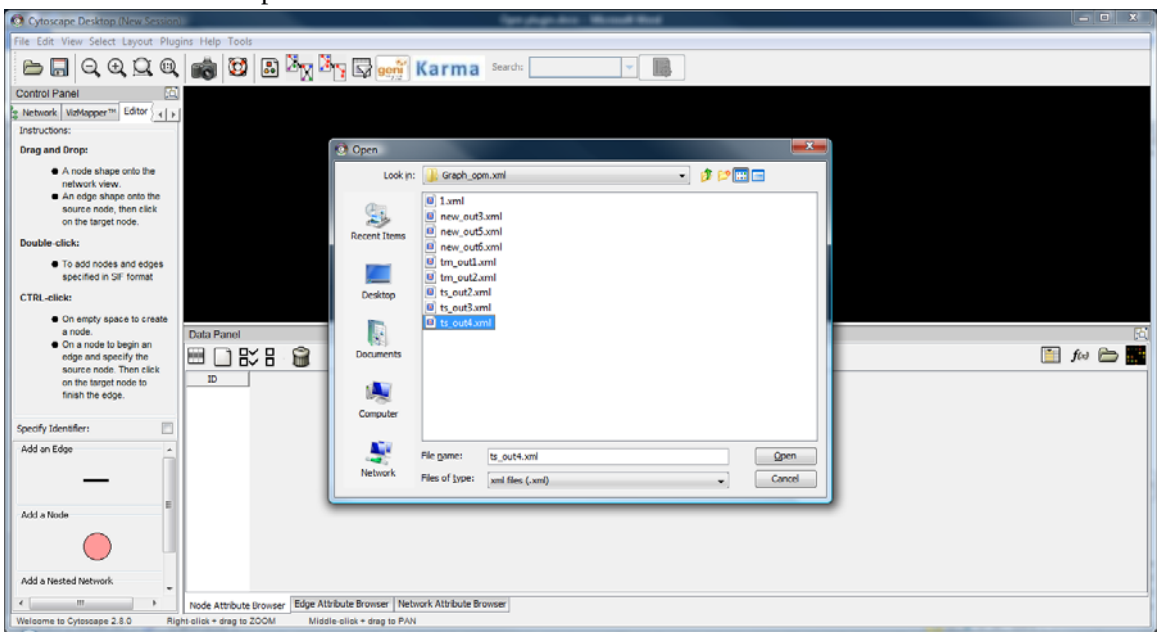

Figure 6.1.3 Select the Karma provenance XML file

4) Load the Karma provenance graph

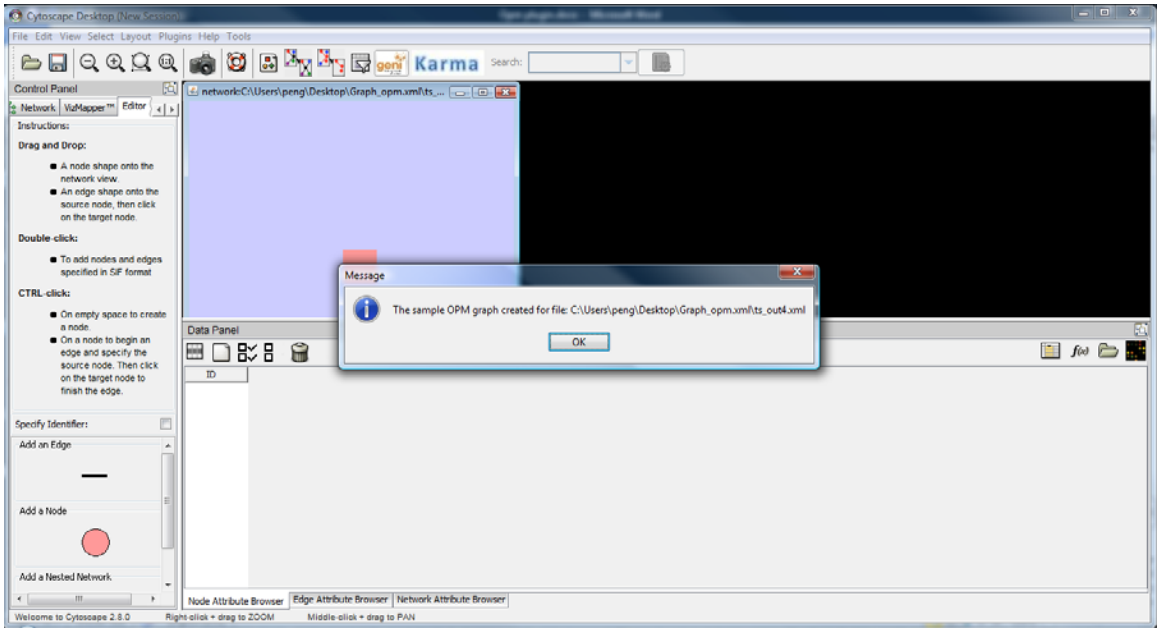
Cytoscape Desktop (New Session)
File Edit View Select Layout Plugins Help Tools
Karma
Search:
Control Panel
Network VizMapper™ Editor
Instructions:
Drag and Drop:
A node shape onto the network view.
An edge shape onto the source node, then click on the target node.
Double-click:
To add nodes and edges specified in SIF format
CTRL-click:
On empty space to create a node.
On a node to begin an edge and specify the source node. Then click on the target node to finish the edge.
Specify Identifier:
Add an Edge
Add a Node
Add a Nested Network
Welcome to Cytoscape 2.8.0
Right-click + drag to ZOOM
Middle-click + drag to PAN
Message
The sample OPM graph created for file: C:\Users\peng\Desktop\Graph_opm.xml\ts_out4.xml
OK
Data Panel
ID
Node Attribute Browser
Edge Attribute Browser
Network Attribute Browser

Figure 6.1.4 Load the Karma provenance graph

The initial graph will look like this (your graph may differ depending on your experiment's provenance):

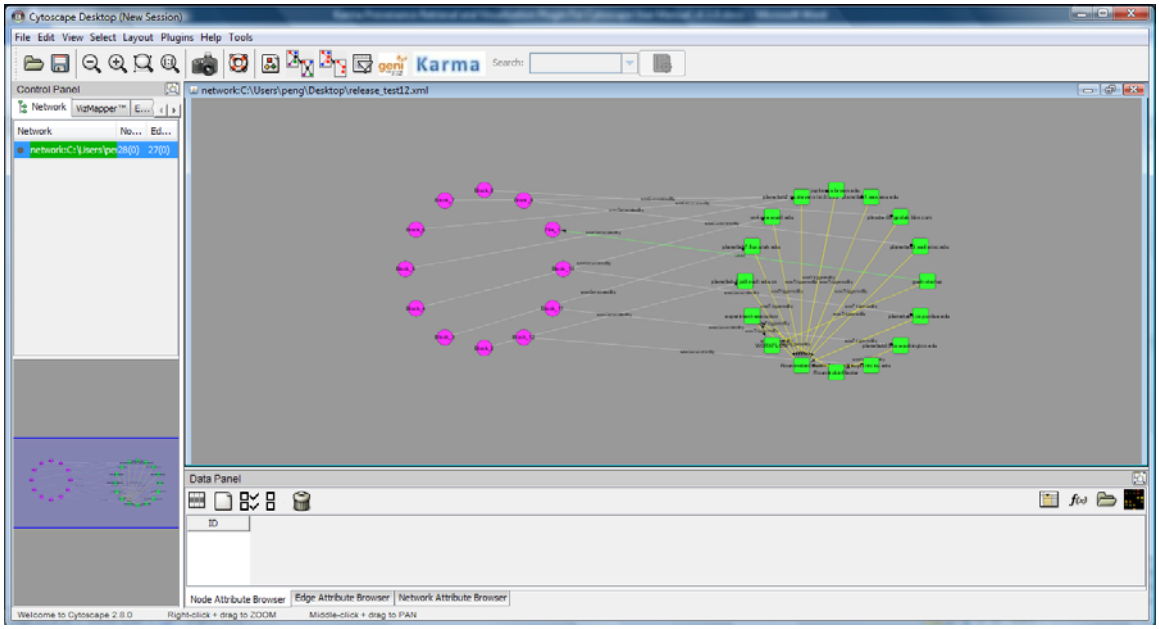

Figure 6.1.5 Initial graph

6.2 Applying layout algorithms to the Karma provenance graph

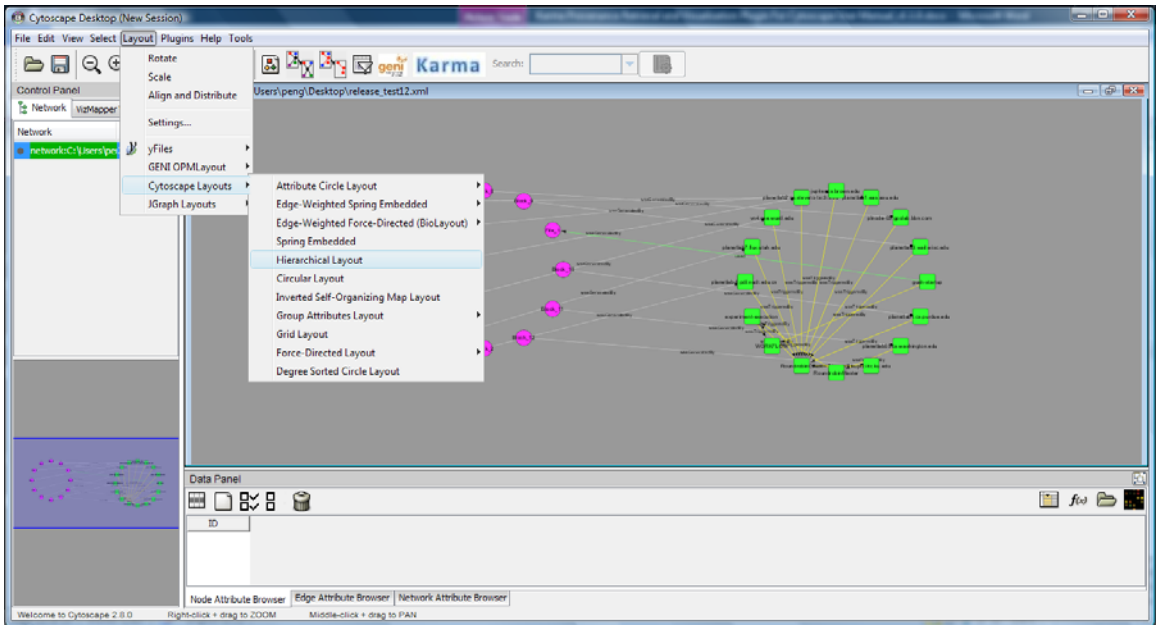

Figure 6.2.1 Do hierarchy layout

1) Different layouts for the provenance graph are available from the Cytoscape Layout/Cytoscape Layouts menu. For example, if the hierarchy layout is selected, the graph will appear as shown in Figure 6.2.2:

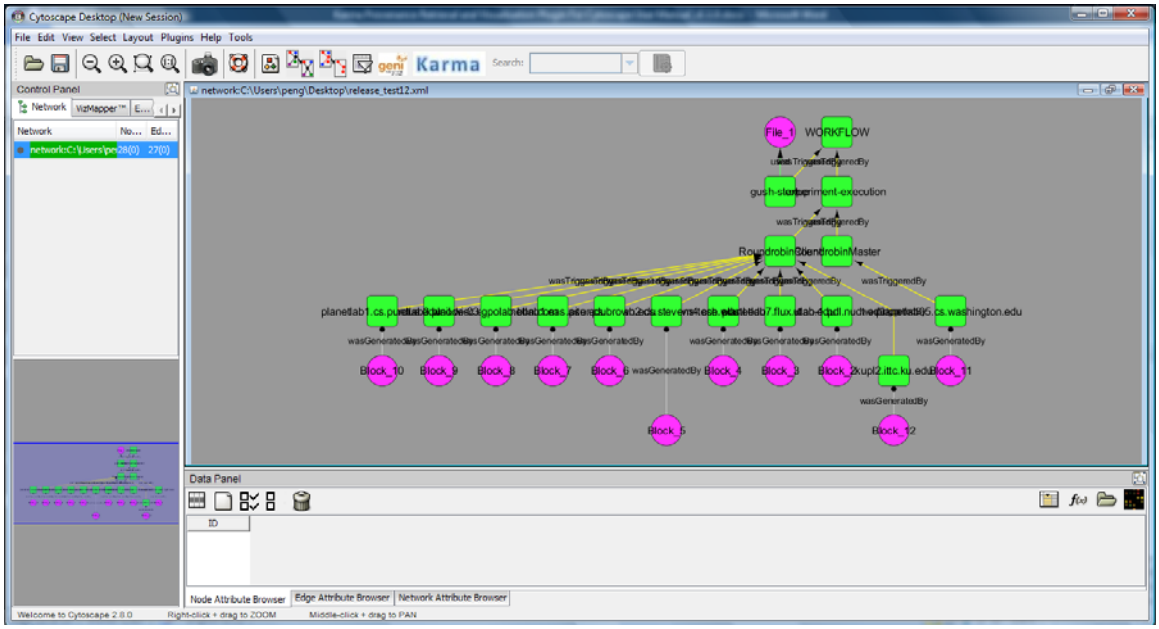

Figure 6.2.2 Hierarchical layout of graph

2) There are several special layout algorithms developed for OPM graphs. Some are designed for general OPM graphs – X axis sort, others are for specific category of OPM graphs – Network Simulator Layout (NS Layout), Network Simulator Layout Extension 1 (NS Layout Ext1). Those will be introduced in the following sections.

6.2.1 X axis sort

The nodes can be displayed using a layered format based on the order of the "time" attribute by selecting the "OPM X sort" option from the Layout/GENI OPMLayout menu.

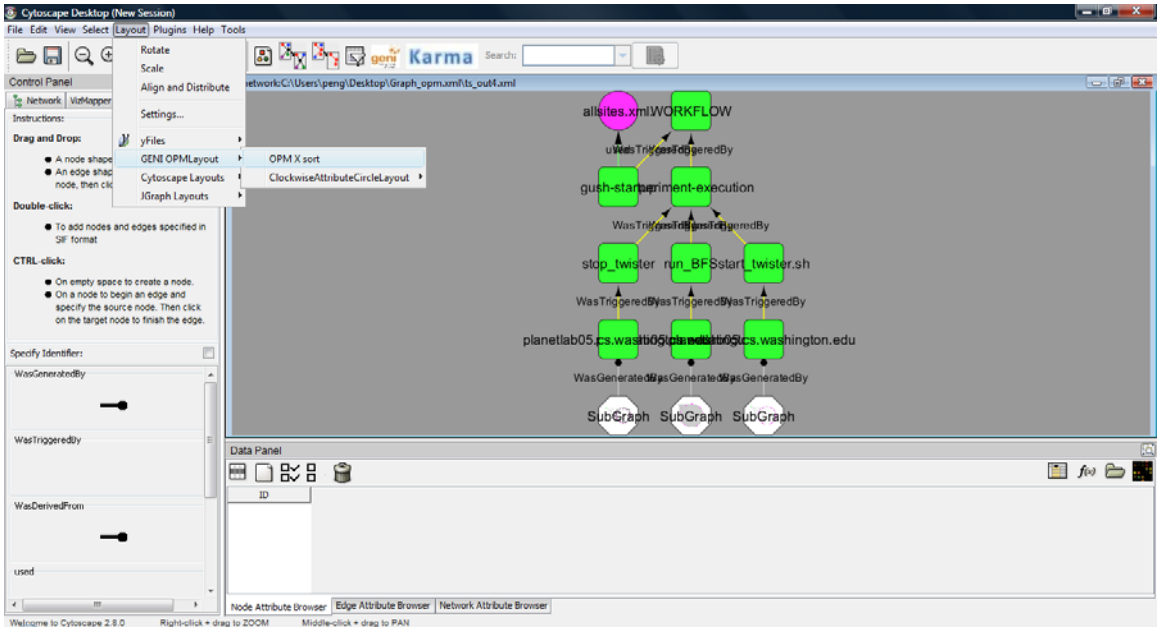

Figure 6.2.1.1 Select the X sort algorithm

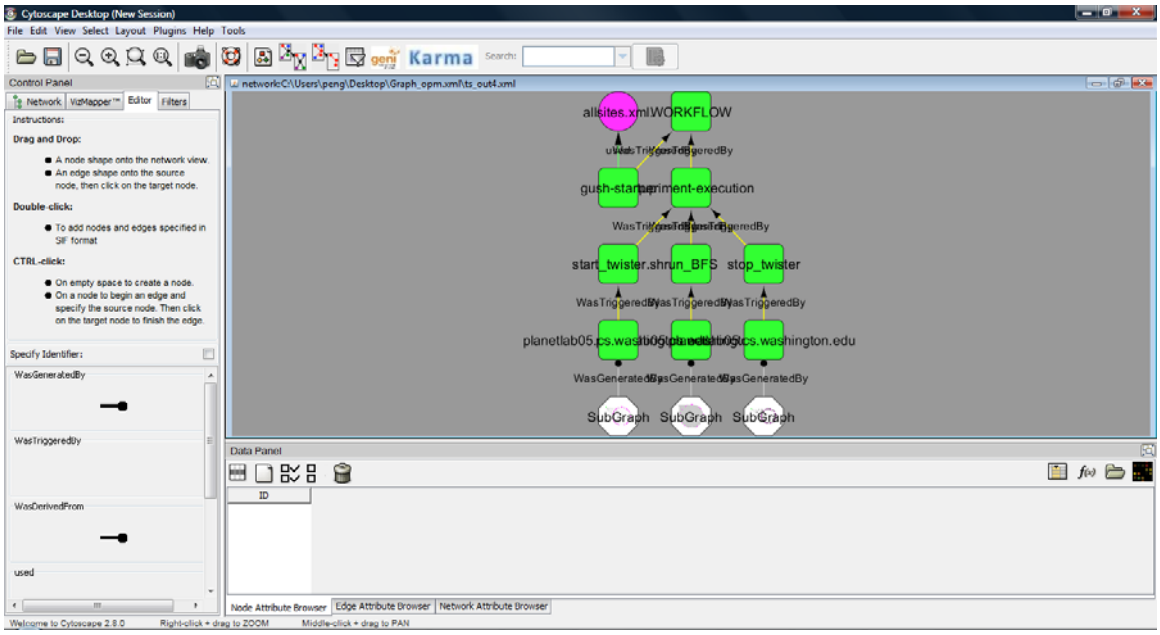

Figure 6.2.1.2 The hierarchy graph after sorting

6.2.2 Network Simulator Layout

The nodes will be positioned based on their category and location coordinates (if they have). That is, for network simulation nodes (OPM PROCESS), they are positioned according to location coordinates; for network events/actions (OPM PROCESS), they are positioned in a small circle surrounding the network simulation node by which they were triggered; for network traffic packets (OPM ARTIFACT), they are positioned in a big circle surrounding the network simulation node by which they were generated.

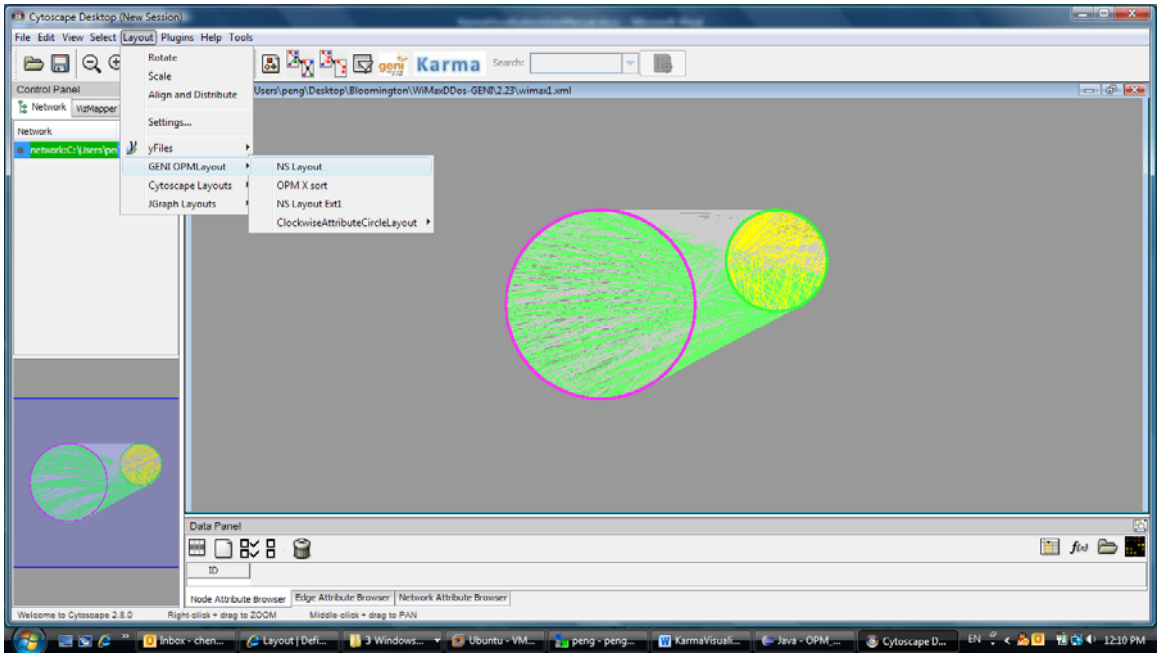

Figure 6.2.2.1 Select the Network Simulator Layout algorithm

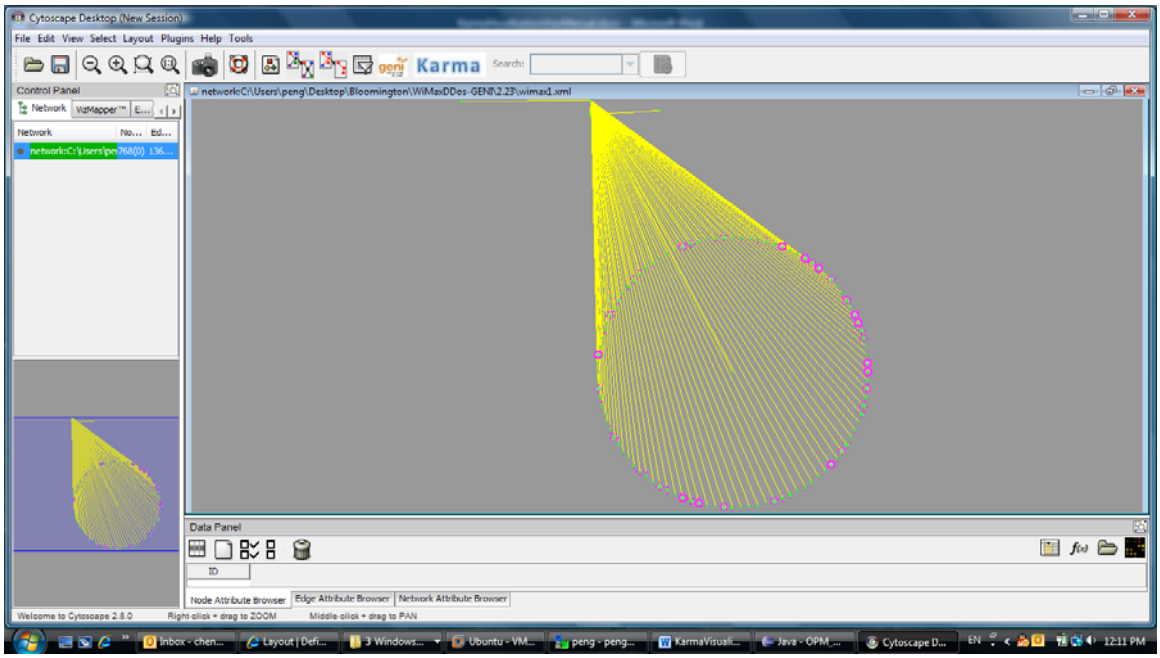

Figure 6.2.2.2 The geo-graph after layout

6.2.3 Network Simulator Layout Extension 1

More than layout, this extension eliminate the network events/actions and network traffic packets from the graph, and visualize the traffic information for individual node in a more abstract and straightforward way. That is, for each network simulation node, it does some statistics on the surrounding packets, and then removes all the packets and events/actions from the graph, displaying the statistic data using the size and graphic of network simulation node instead.

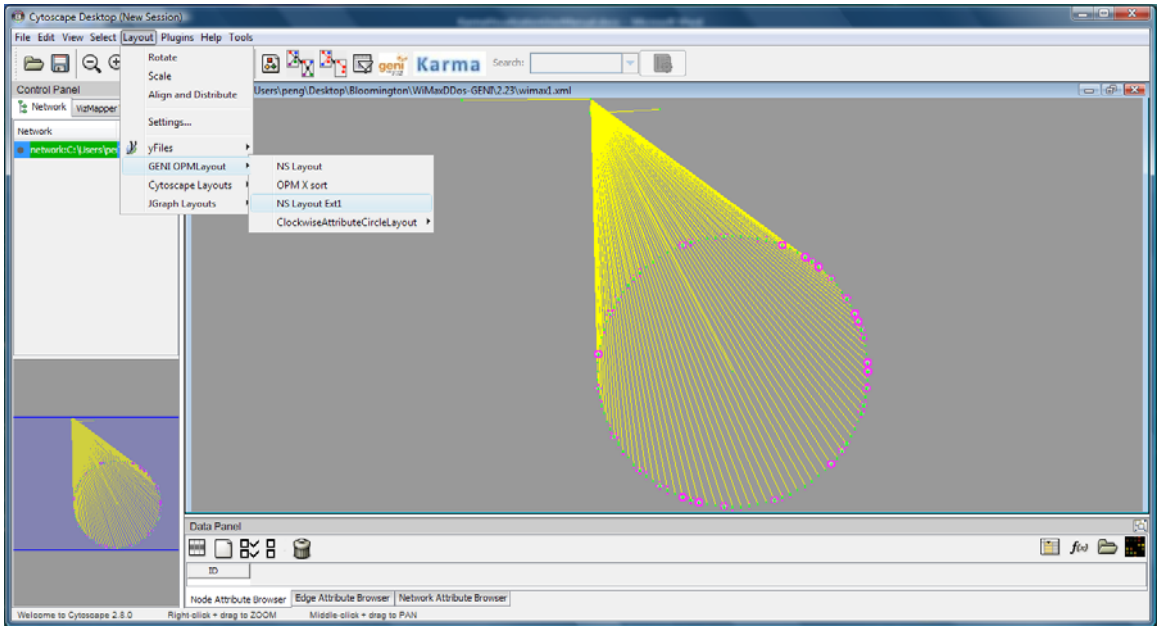

Figure 6.2.3.1 Select the Network Simulator Layout Extension 1

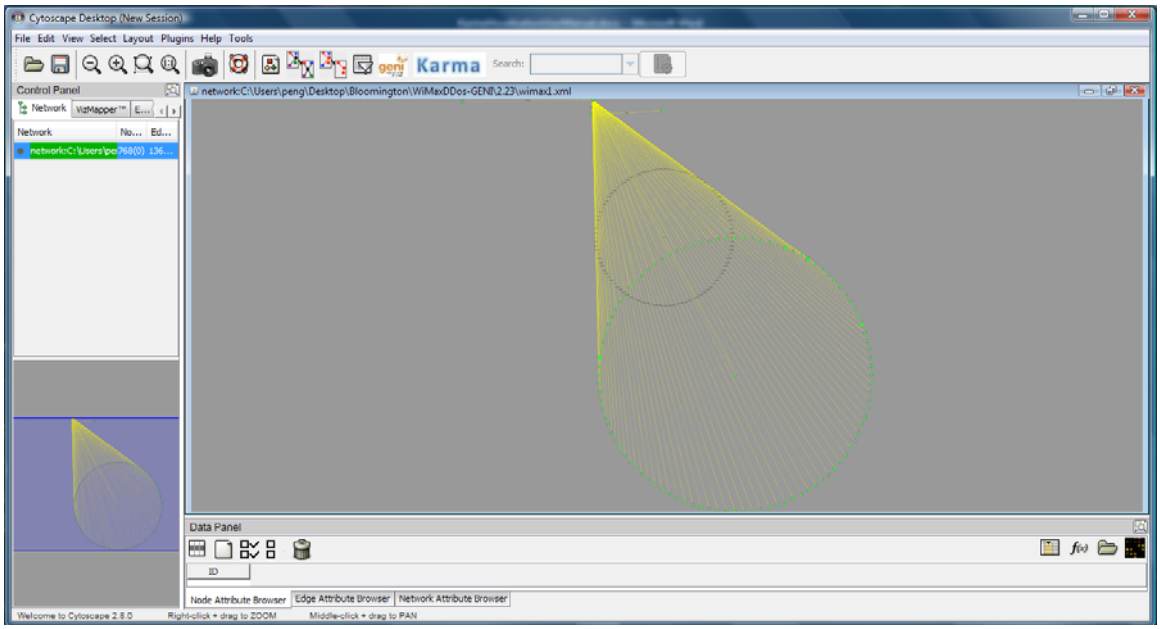

Figure 6.2.3.2 The abstract graph after layout. The size of each node indicates the number of surrounding packets– bigger node size indicates more surrounding packets.

User can use customized node graphics to show the statistics on traffic packets:

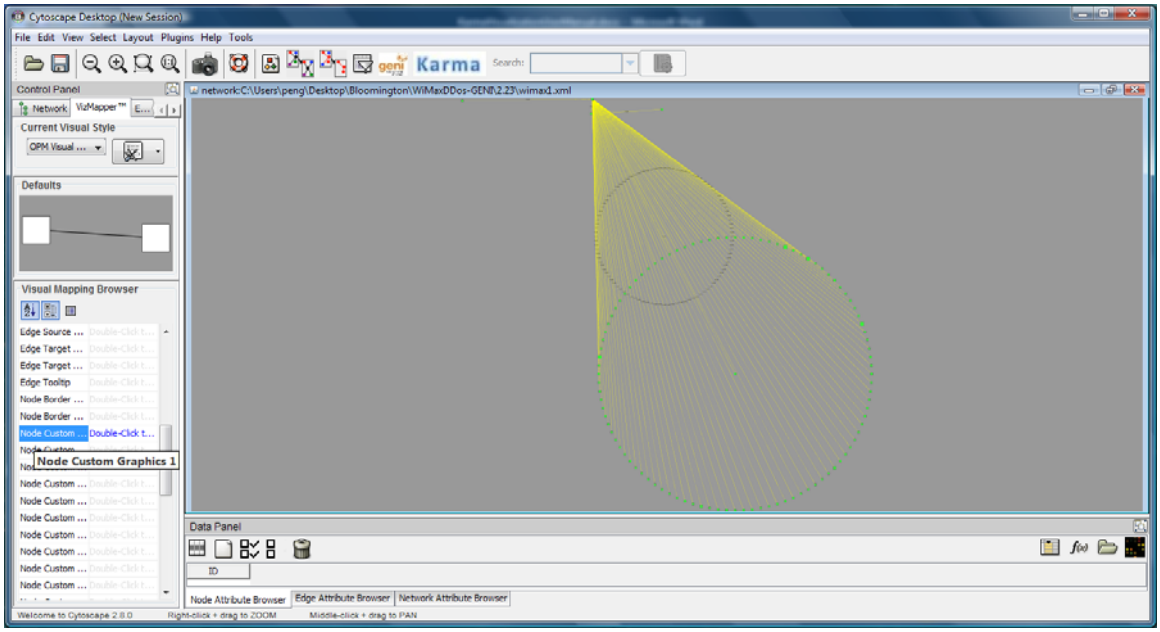

Figure 6.2.3.3 Add the Node Customized Graphics visual mapping

To show the statistics on dropped&sent packets, user can map the graphics to “graph-url-drop&sent”:

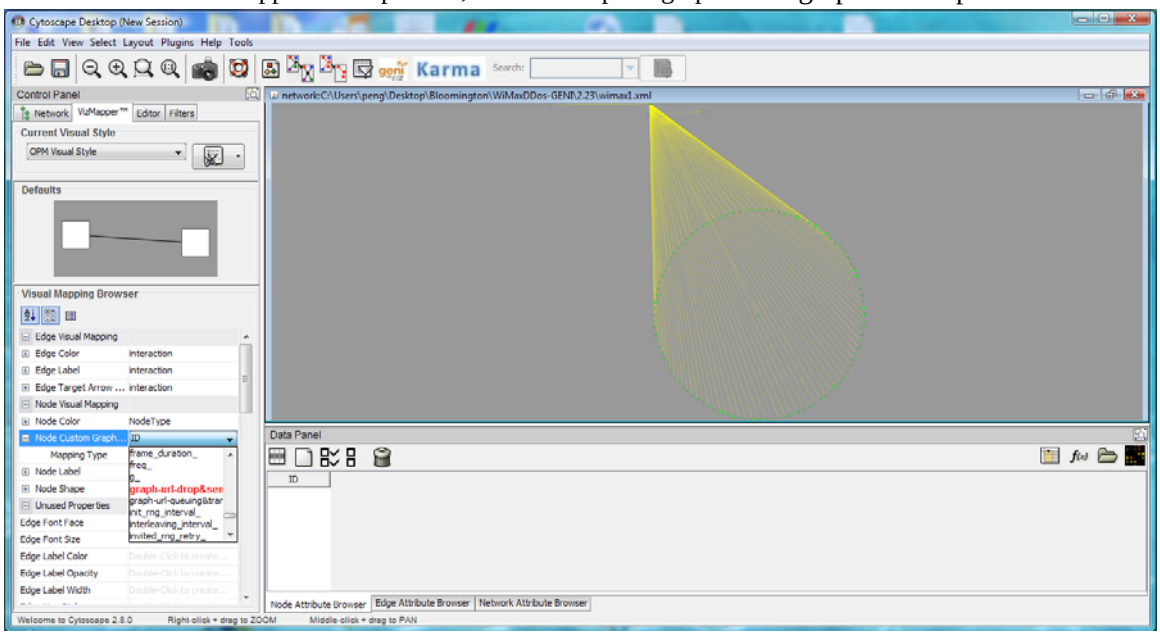

Figure 6.2.3.4 Set the Node Customized Graphics visual mapping to “graph-url-drop&sent”

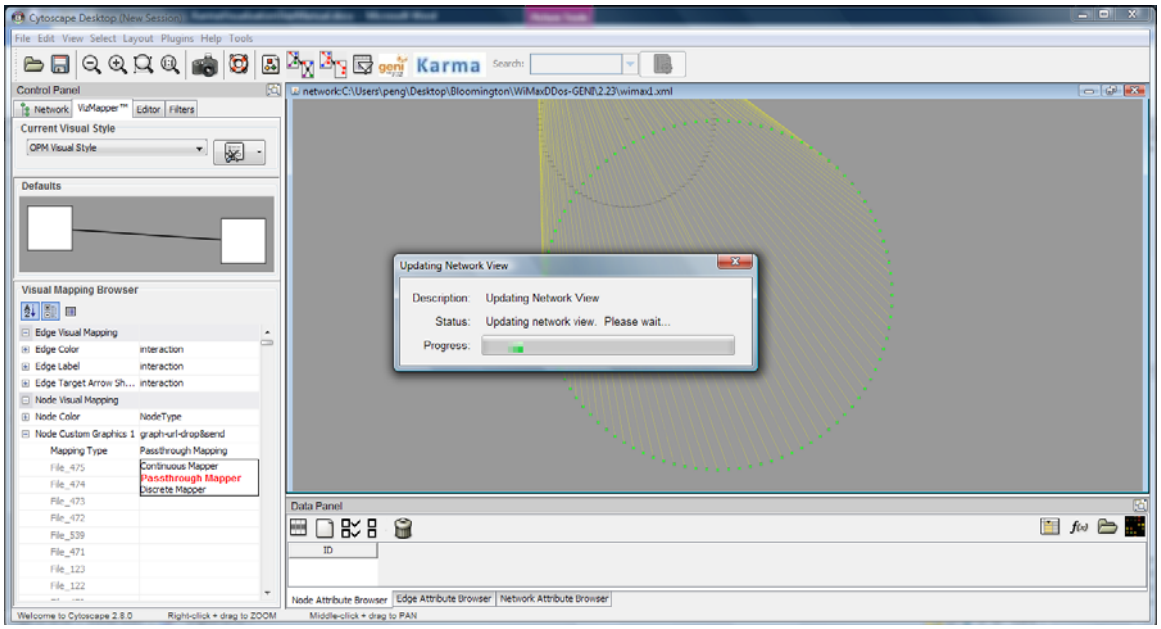

Figure 6.2.3.5 Set the mapping type to “Passthrough Mapper”

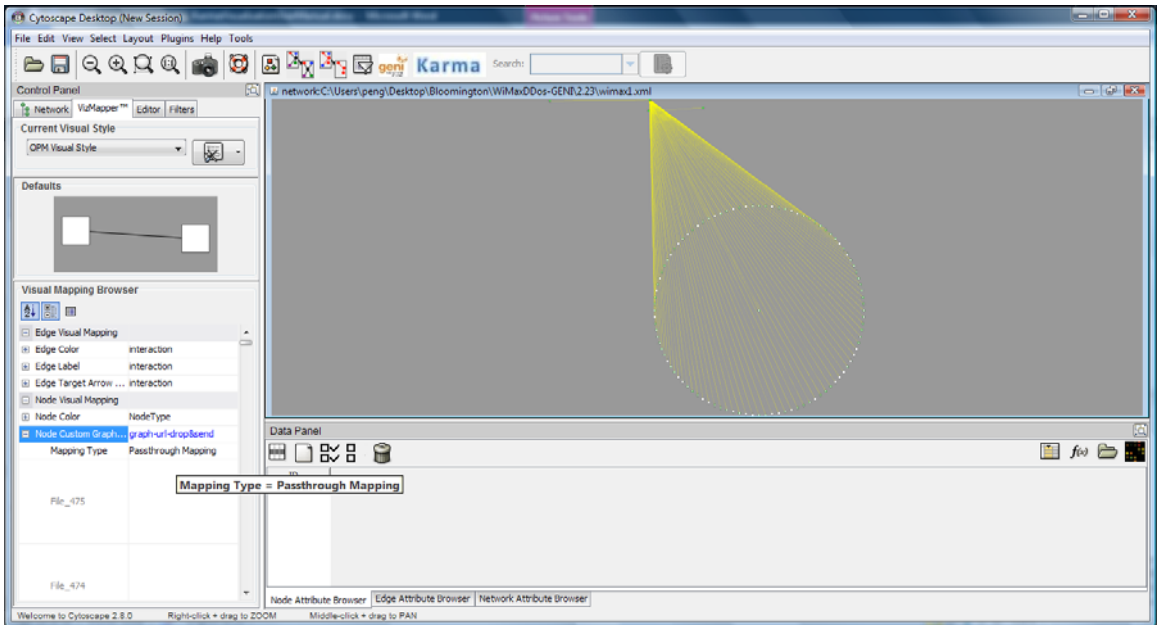

Figure 6.2.3.6 Graph with customized node graphics showing the statistic of packets been dropped and sent

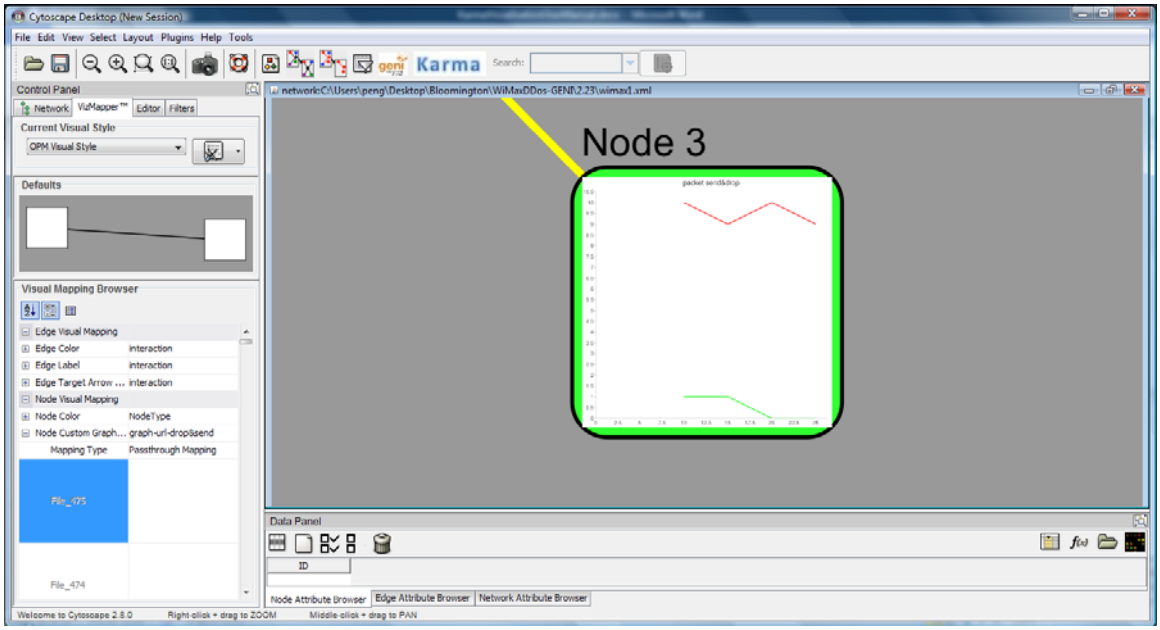

Figure 6.2.3.7 Customized node graphic showing the statistic of packets been dropped and sent. Blue line shows the number of packets dropped every 5 seconds, and red line represents the number of packets sent every 5 seconds.

To show the statistics on packets queuing time and pakcets transfer time, user can map the graphics to "graph-url-queuing&transfer":

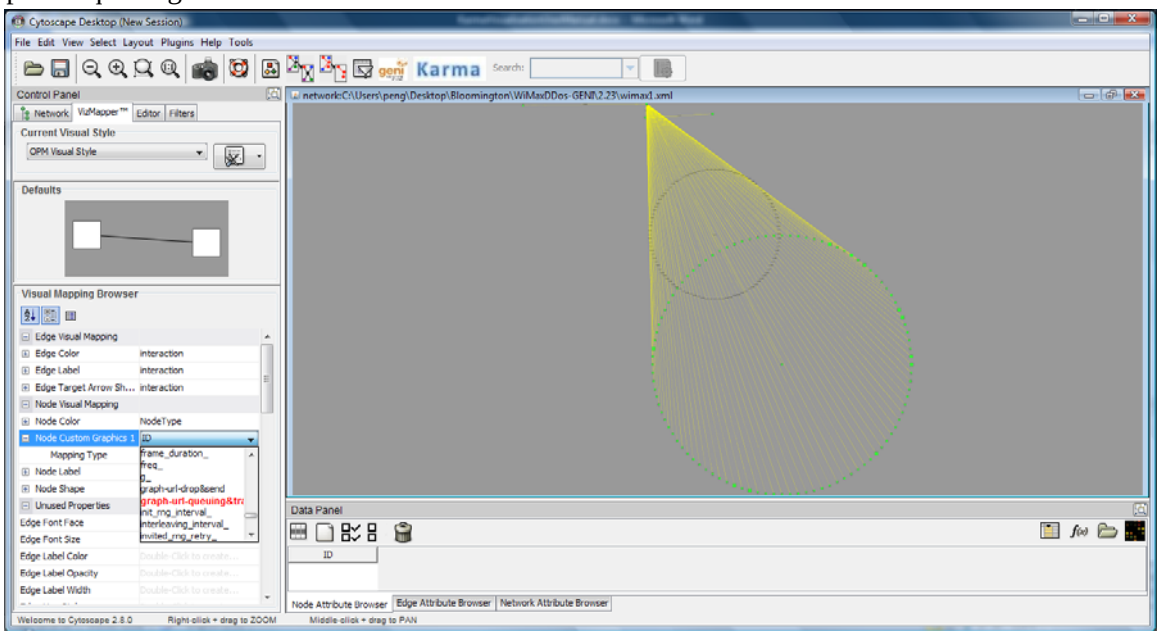

Figure 6.2.3.8 Set the Node Customized Graphics visual mapping to "graph-url-queuing&transfer"

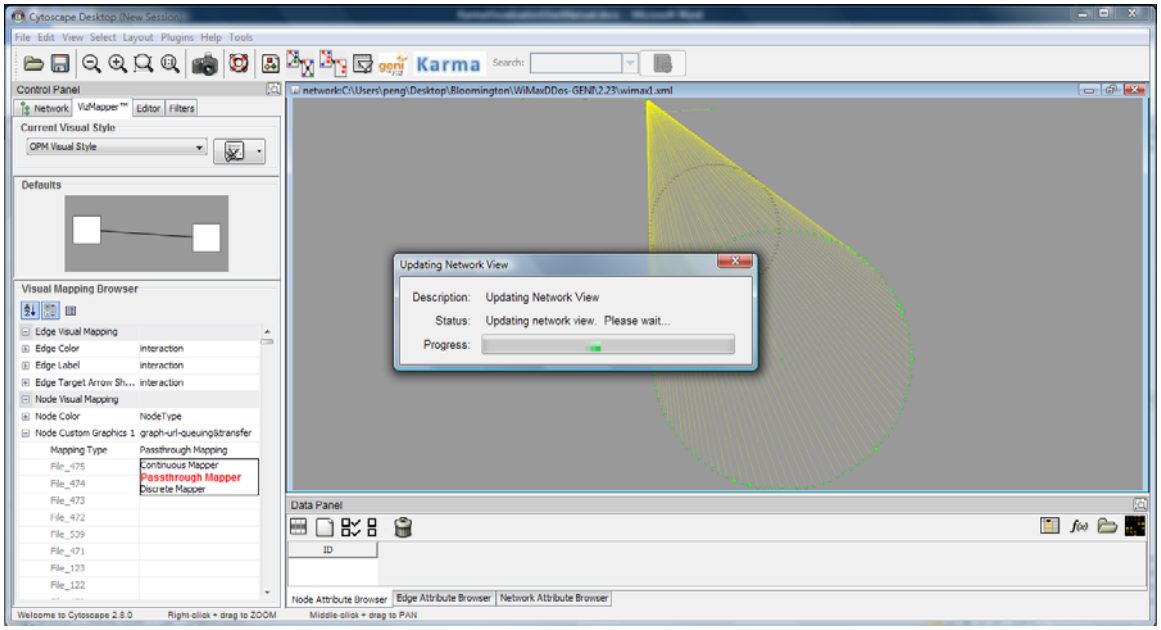

Figure 6.2.3.9 Set the mapping type to "Passthrough Mapper"

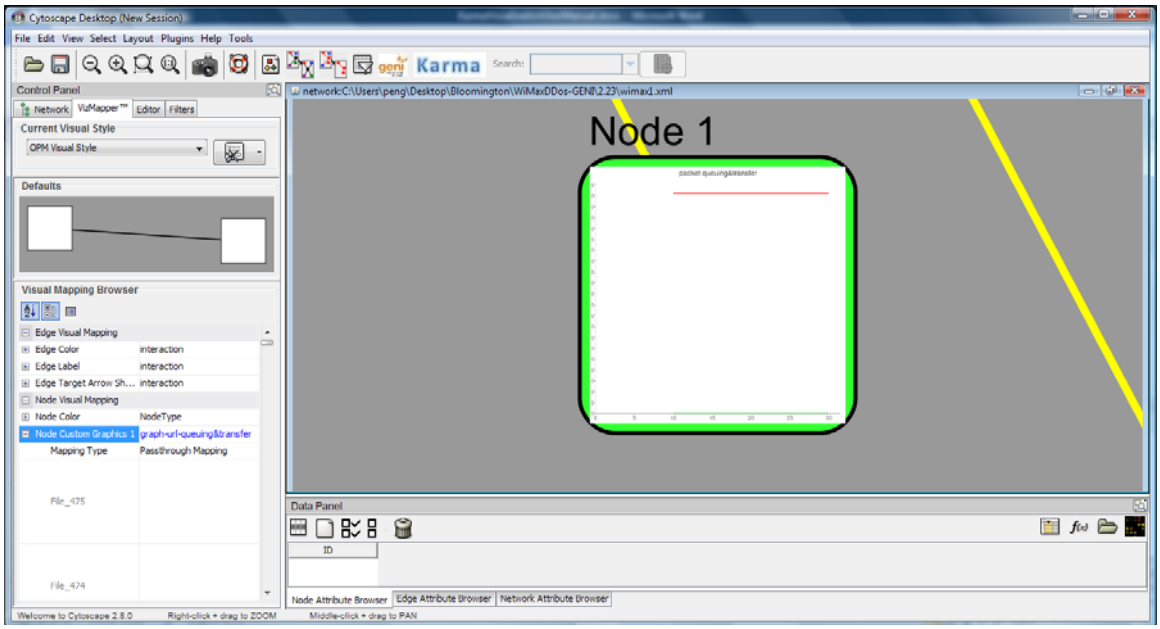

Figure 6.2.3.10 Customized node graphic showing the statistic of average packet queuing time and packet transfer time. Blue line shows the average packet queuing time, and red line represents average packet transfer time.

6.3 Navigating the Karma provenance graph

1) Double click on any node to see its attributes

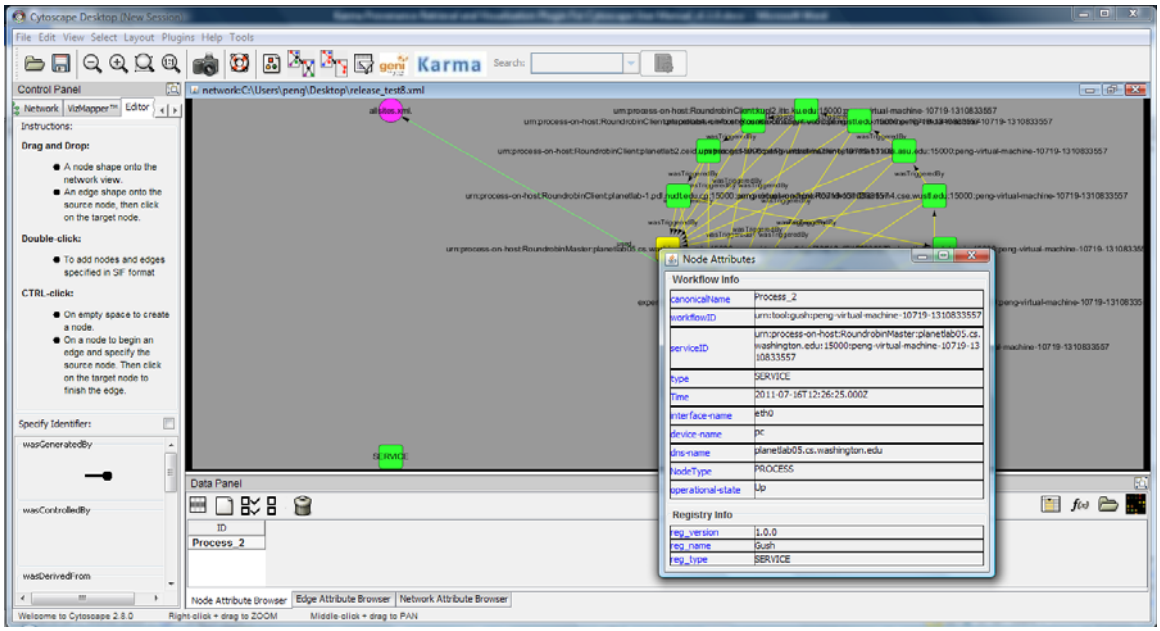

Figure 6.3.1 Double click on node

2) Double click on any edge to see its attributes

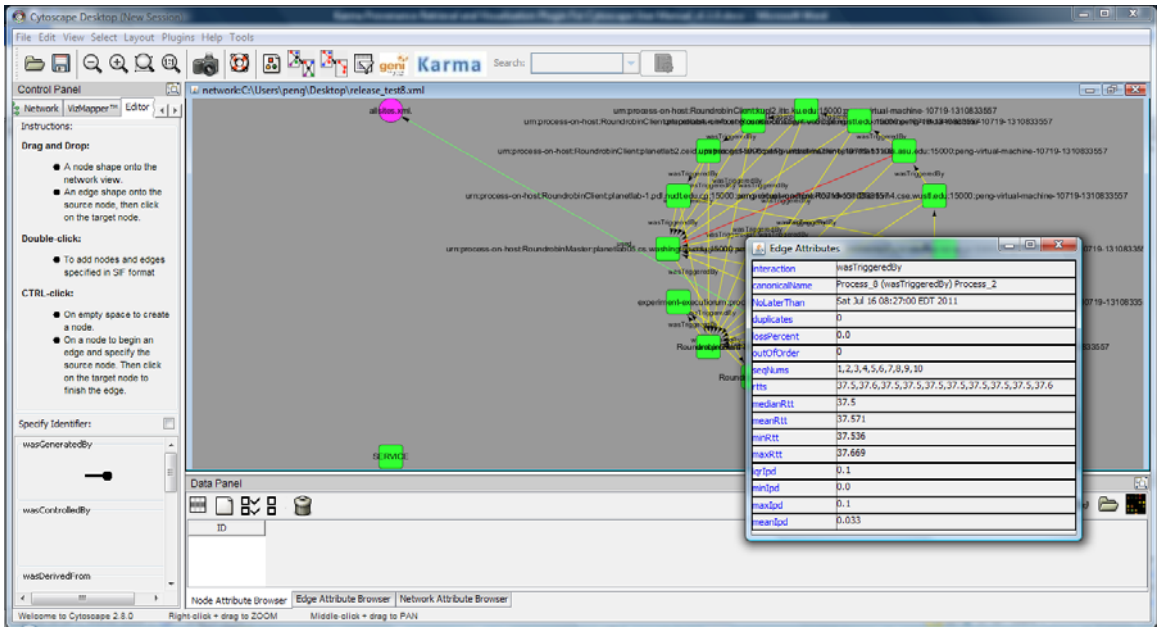

Figure 6.3.2 Double click on edge

6.4 Playing movie

If any node is right clicked, a movie displaying the provenance for the entire graph can be played. The movie is based on the order of the “timestep” or “time” attributes of the nodes.

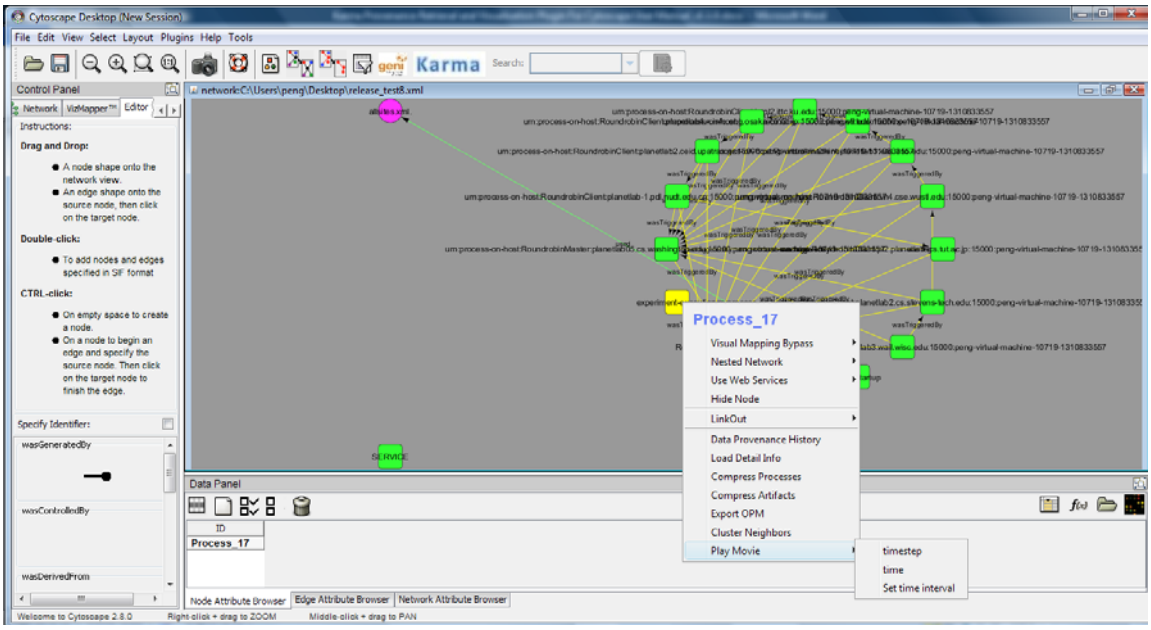

Figure 6.4.1 Display provenance as a movie

The time interval (on milliseconds) can be configured between the appearances of two nodes in the movie.

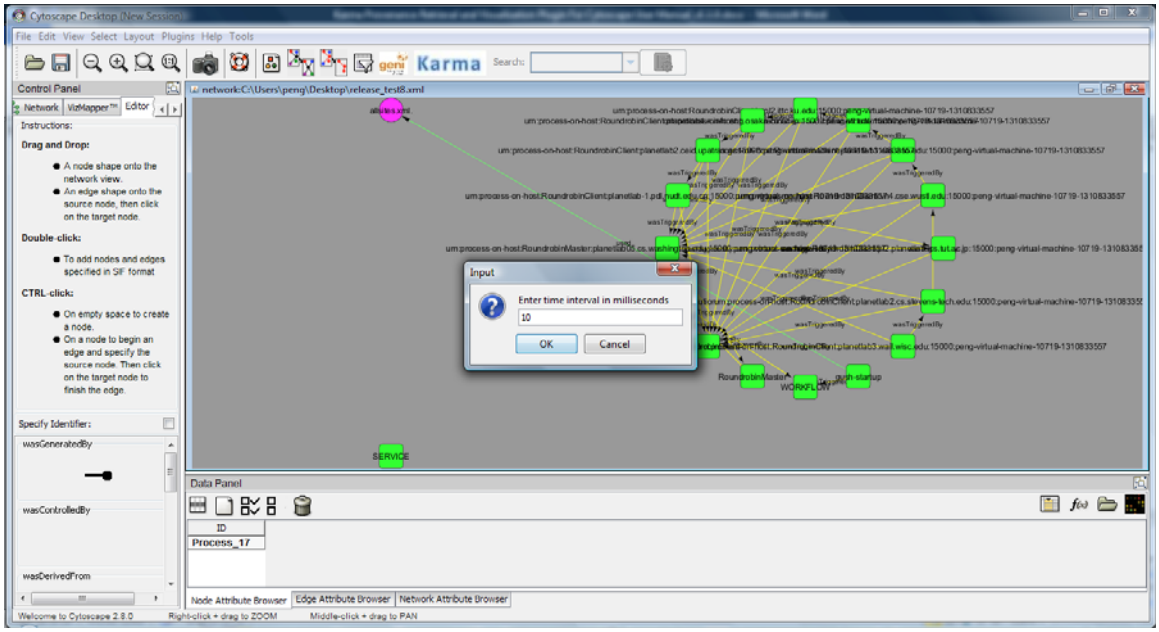

Figure 6.4.2 Configure the time interval

6.5 Saving graph as XML

To save the displayed provenance in Cytoscape as an XML file that is based on the Open Provenance Model (OPM), right click on any node and select the menu item “Export OPM”. The user will be prompted for the location to save the XML file.

The graph’s structure will be based on the OPM v1.1 (Open Provenance Model, http://openprovenance.org/).

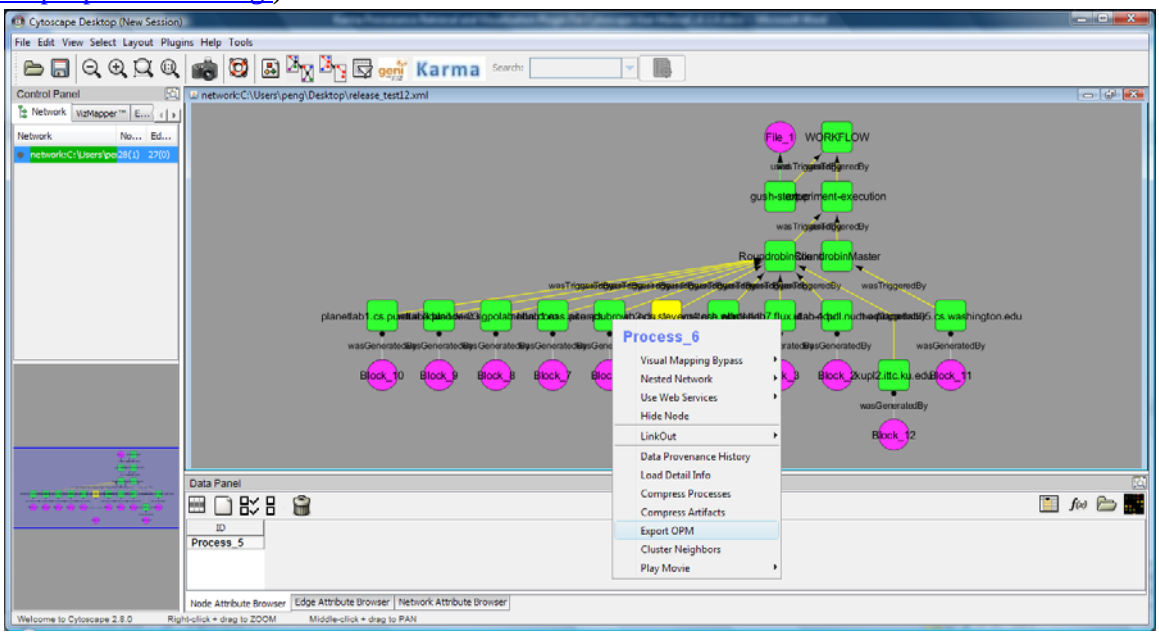

Figure 6.5.1 Click on “Export OPM”

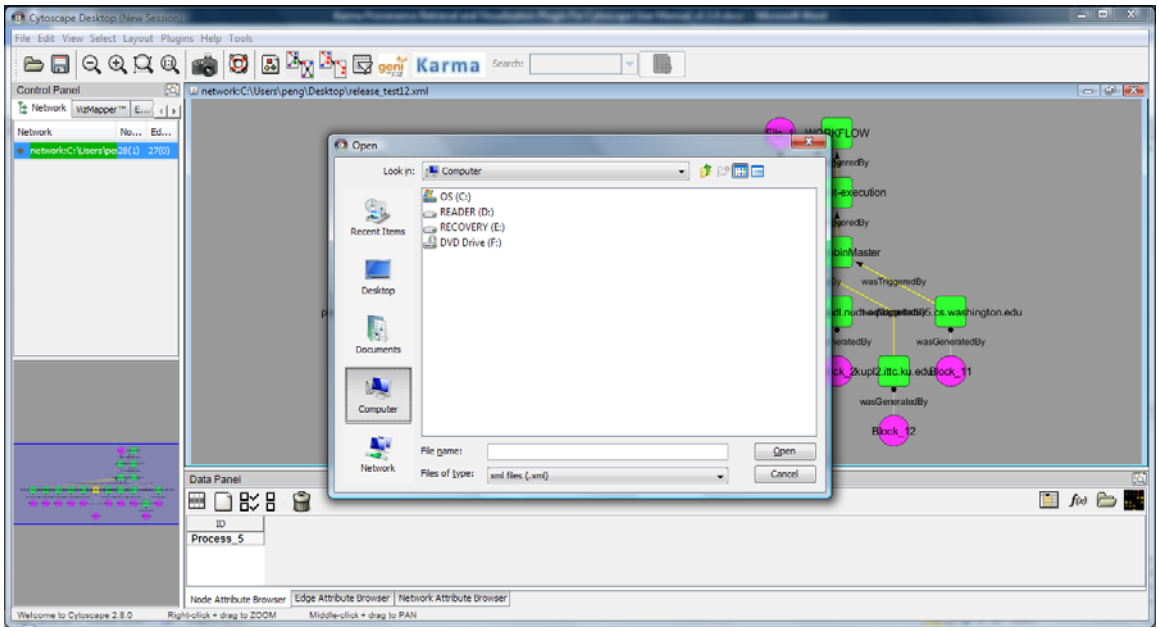

Figure 6.5.2 Choose the location to save the xml file

6.6 Getting data provenance history

Right clicking on any artifact will display an option for “Data Provenance History”. Selecting that option will display a new graph that includes all of the nodes involved in the generation of the selected artifact (This method invokes background communications with the Karma server).

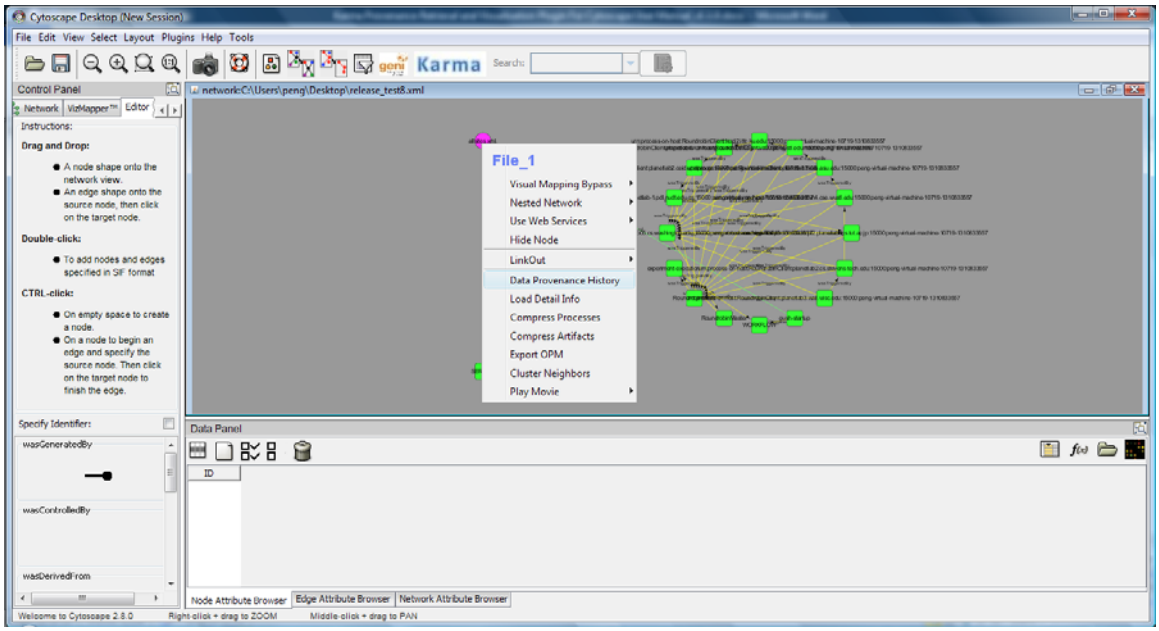

Figure 6.6.1 Get data provenance history

Note: This functionality works best when the Karma visualization plugin is connected to Karma using Axis2 webservice instead of the RabbitMQ.

6.7 Creating visual style

User can create customized visual style through VizMapper control panel in Cytoscape. We also developed a special visual style for Network Simulation graph showing WiMax DDoS experiment.

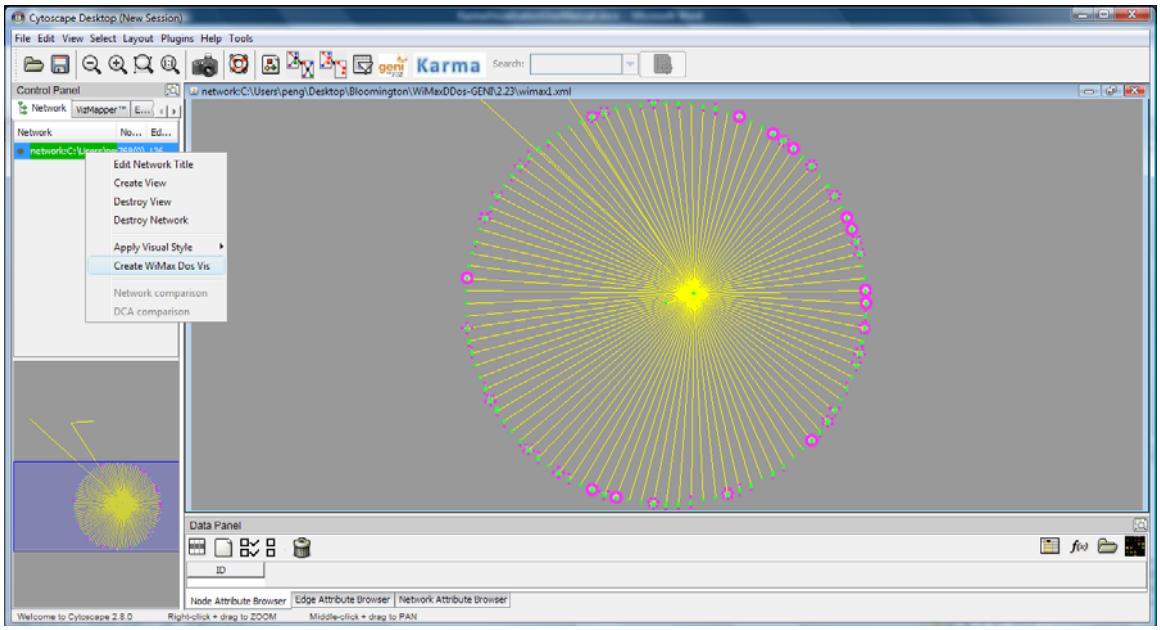

Figure 6.7.1 Right click on the network in the network control panel, and then select the “Create WiMax DoS Vis”.

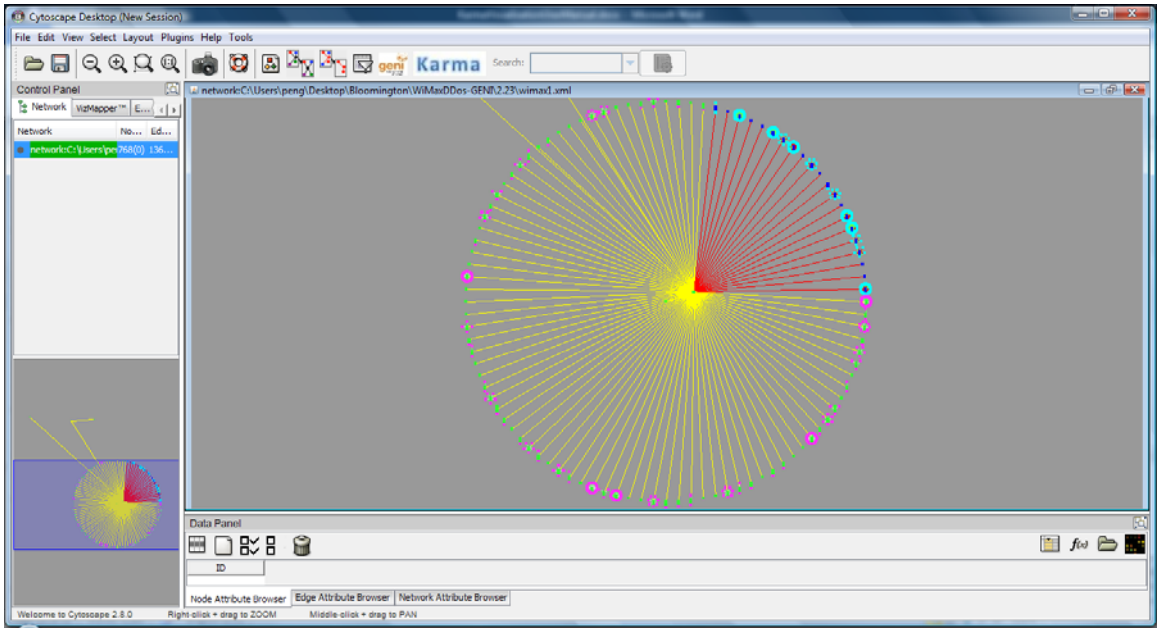

Figure 6.7.2 WiMax DDoS provenance graph using different colors in differentiating attackers and users. The attackers and their action/event nodes are marked in blue, edges connected to them are marked in red, and the traffic packets surrounding attackers are marked in Cyan.

6.8 Creating an abstract view

For some experiments, the complexity of the provenance relationships can result in very complicated graphs. This section introduces features of the Karma visualization plug-in that can be used to abstract out some of this complexity to allow visualizing specific aspects of the provenance graph.

6.8.1 Clustering neighbor nodes

To deal with graphs with a large number of "artifact" nodes, the Karma visualization plugin supports extracting an abstract view by clustering neighboring nodes. Figure 6.8.1.1 shows an example of a graph that has a larger number of "artifact" nodes generated by a small number of "Process" nodes.

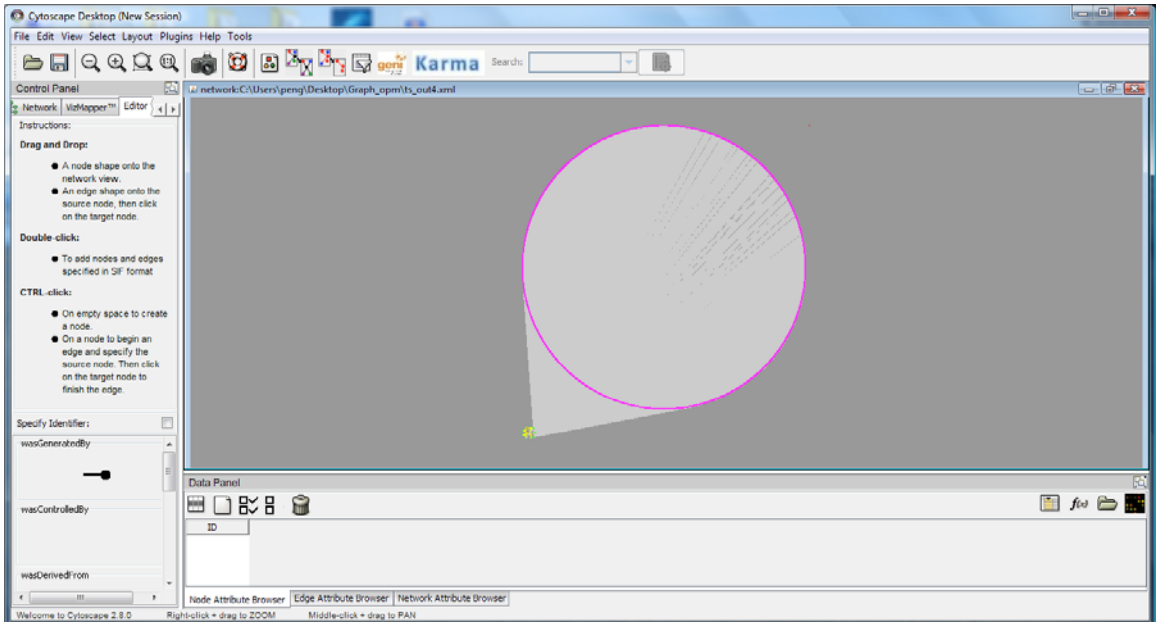
Cytoscape Desktop (New Session)
File Edit View Select Layout Plugins Help Tools
Karma
Control Panel
Network
VizMapper™
Editor
Instructions:
Drag and Drop:
A node shape onto the network view.
An edge shape onto the source node, then click on the target node.
Double-click:
To add nodes and edges specified in SIF format
CTRL-click:
On empty space to create a node.
On a node to begin an edge and specify the source node. Then click on the target node to finish the edge.
Specify Identifier:
wasGeneratedBy
wasControlledBy
wasDerivedFrom
Data Panel
ID
Node Attribute Browser
Edge Attribute Browser
Network Attribute Browser
Welcome to Cytoscape 2.8.0
Right-click + drag to ZOOM
Middle-click + drag to PAN

Figure 6.8.1.1 A graph with large number of "Artifact" nodes

The user can create an abstract view for this type of graph using the following 5-step process:

1) Select the process nodes (on the lower left side of the graph in Figure 6.8.1.2):

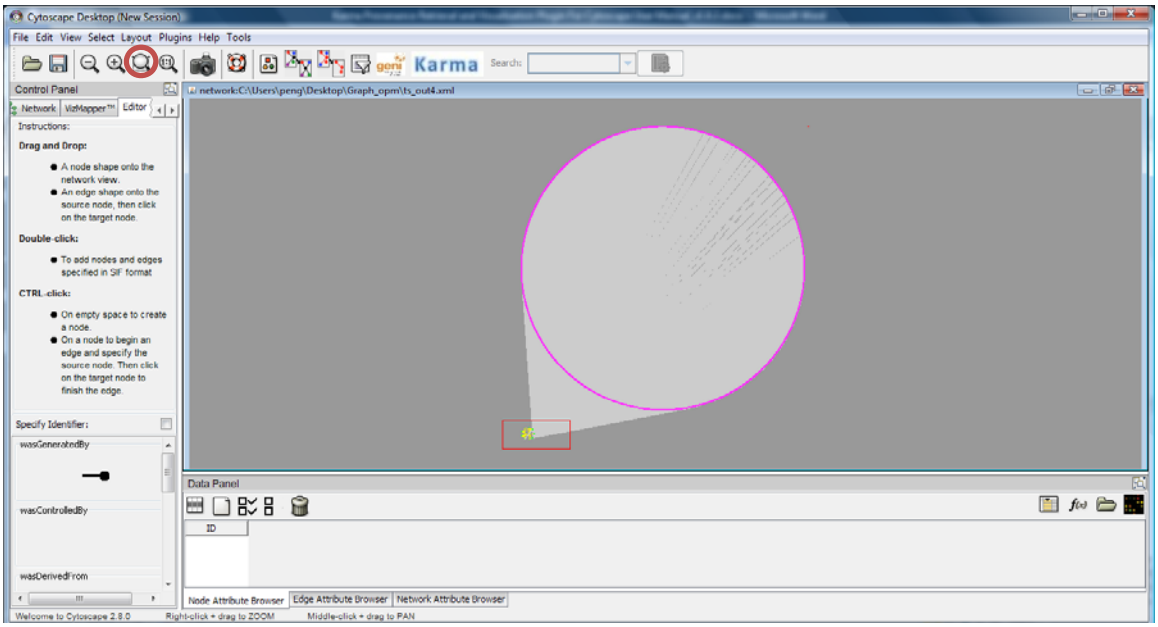
Cytoscape Desktop (New Session)
File Edit View Select Layout Plugins Help Tools
Karma
Control Panel
Network VizMapper™ Editor
Instructions:
Drag and Drop:
A node shape onto the network view.
An edge shape onto the source node, then click on the target node.
Double-click:
To add nodes and edges specified in SIF format
CTRL-click:
On empty space to create a node.
On a node to begin an edge and specify the source node. Then click on the target node to finish the edge.
Specify Identifier:
wasGeneratedBy
wasControlledBy
wasDerivedFrom
Data Panel
ID
Node Attribute Browser
Edge Attribute Browser
Network Attribute Browser
Welcome to Cytoscape 2.8.0
Right-click + drag to ZOOM
Middle-click + drag to PAN

Figure 6.8.1.2 Select processes

2) Click “zoom selected region” icon from the toolbar (highlighted magnifying glass icon the Figure 6.8.1.3):

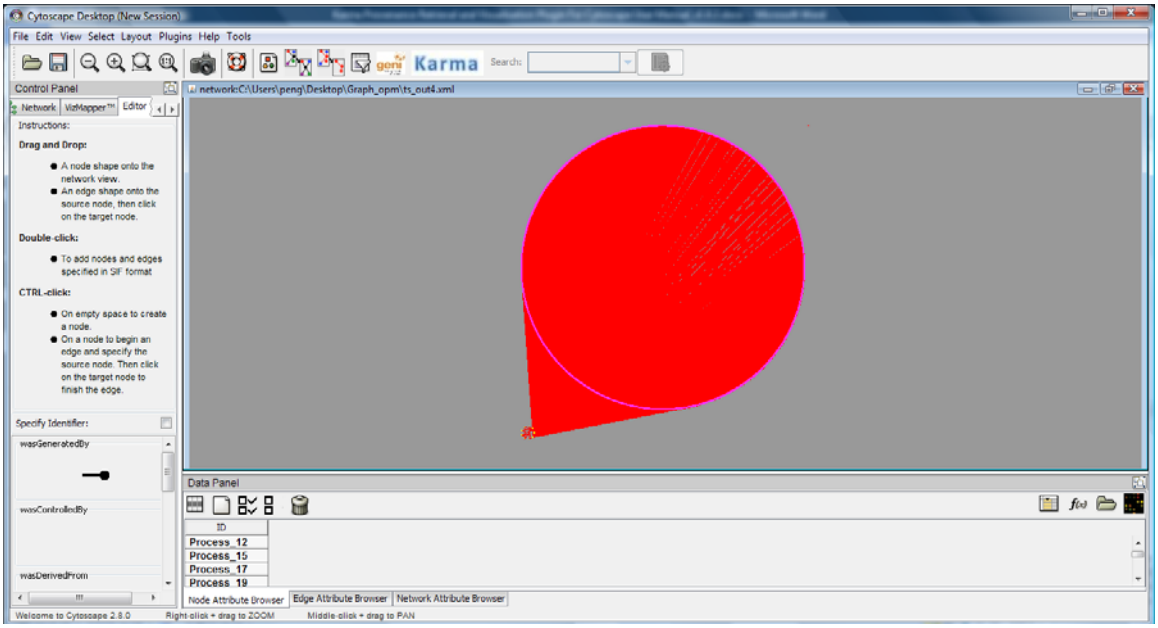

Figure 6.8.1.3 Zoom into process region

3) Right click on one of the nodes with a large out-degree (a significant number of edges):

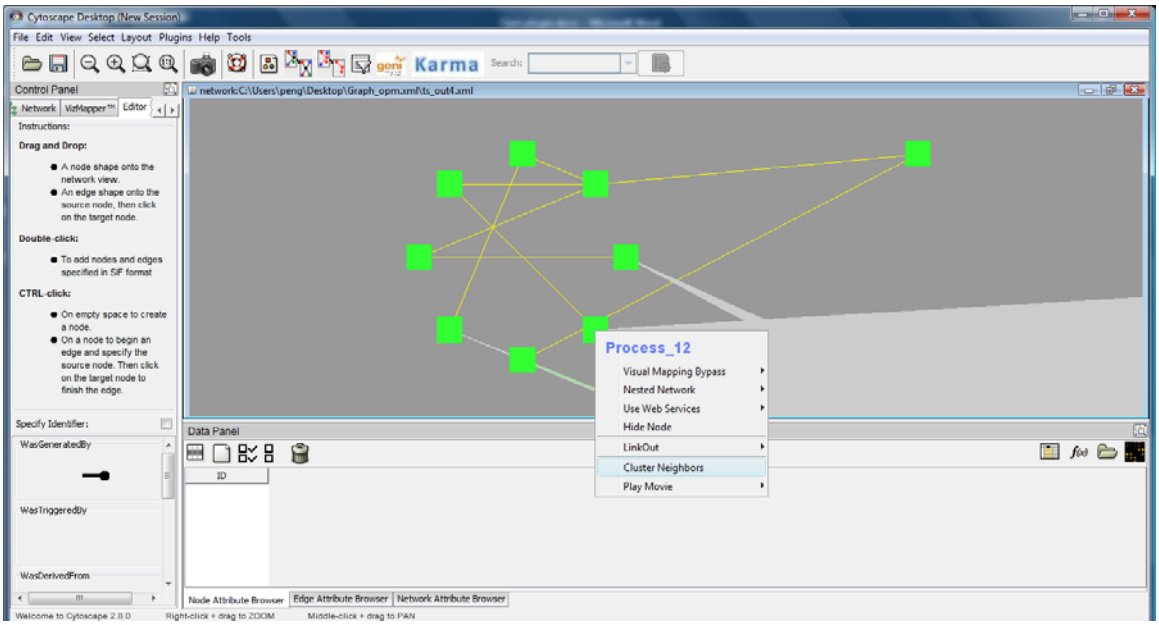

Figure 6.8.1.4 Right click on node

4) Clustering the neighbors of the node selected in step 3:

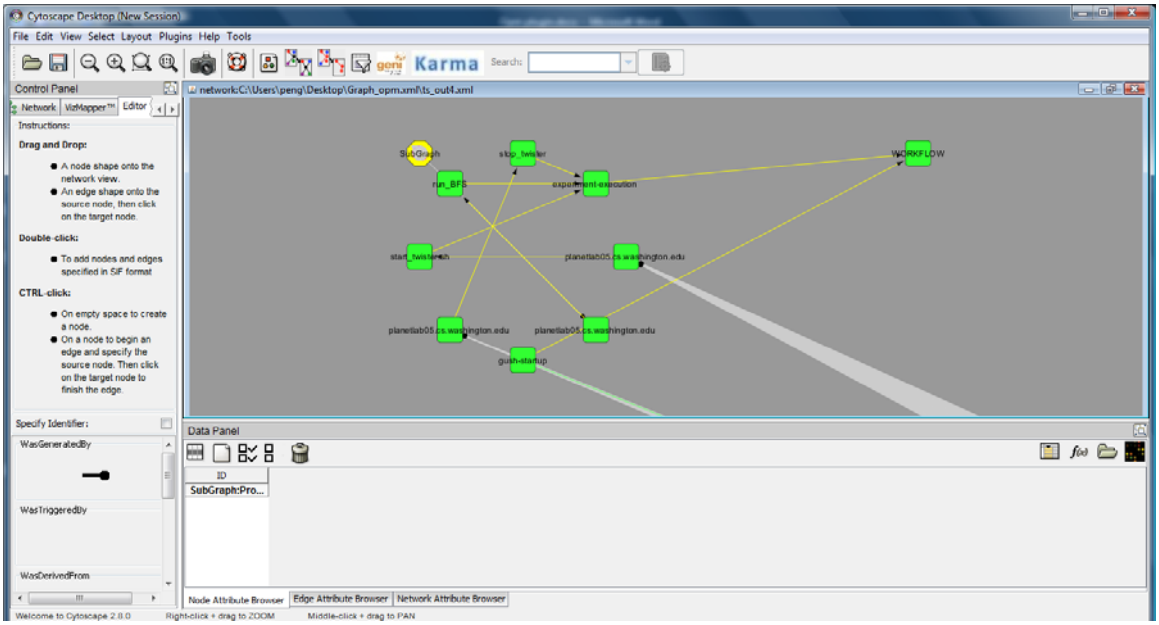

Figure 6.8.1.5 Cluster nodes

5) Repeat step 4 until an abstract graph is obtained showing only the smaller set of nodes that the user intends to visualize.

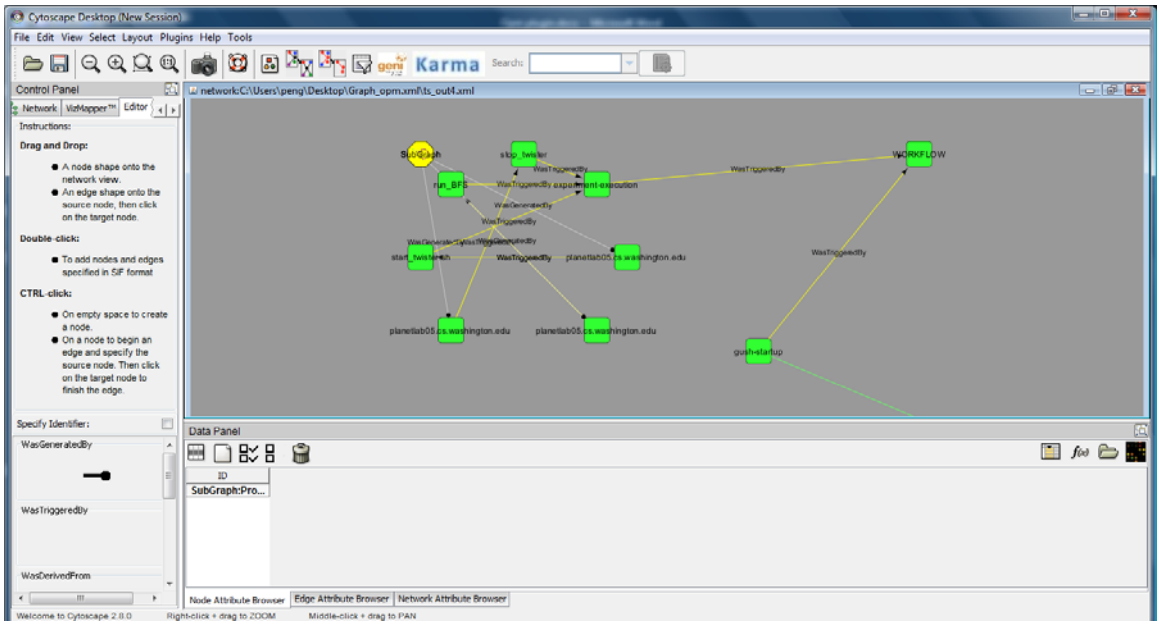

Figure 6.8.1.6 Create abstract graph

6.8.2 Navigate between graph and sub-graphs

Once an abstract view is created, one can navigate between the parent abstract view and the sub-graph for any collapsed node.

1) Double clicking on any abstract node will show the view of the collapsed sub-graph

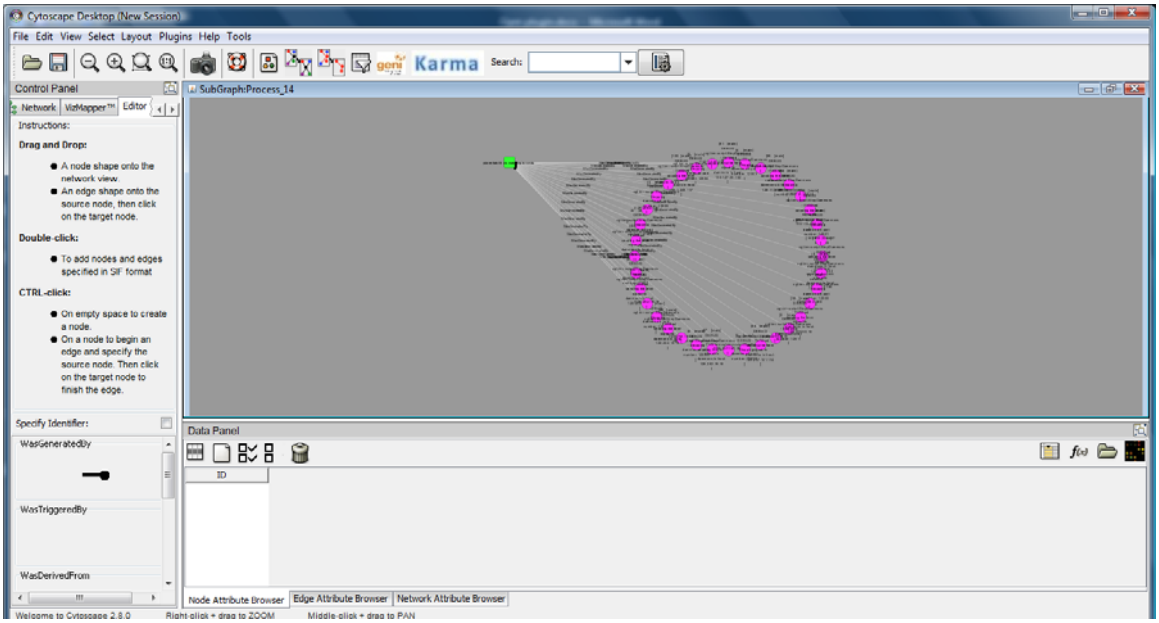

Figure 6.8.2.1 The sub-graph

2) By clicking on the item in the network panel on the left-hand side of the screen, one can go back to the parent graph

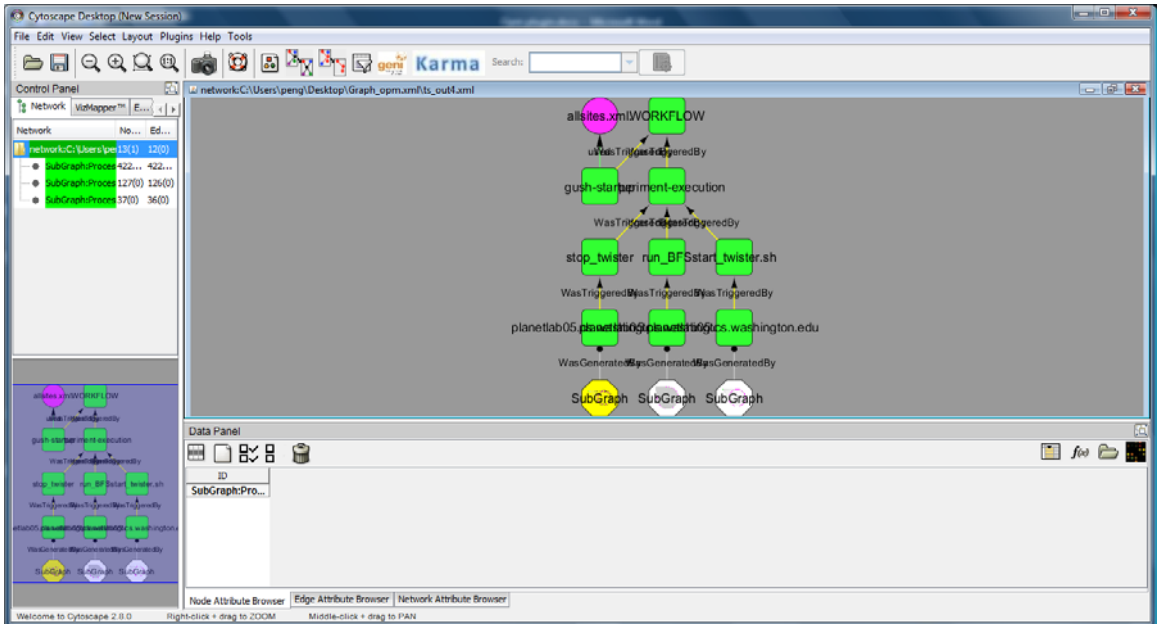

Figure 6.8.2.2 The parent graph

6.8.3 Compress Process/Artifact

Right clicking on any node in the graph will show the menu options for “Compress Process” and “Compress Artifact”.

Clicking on “Compress Process”, the Cytoscape visualization plugin will eliminate all of the “Process” nodes that links two “artifact” nodes with the outgoing edge representing the relationship “used” and the incoming edge representing the relationship “wasGeneratedBy”. The process node will be replaced by a new edge between these two “artifact” nodes that represents the relationship “wasDerivedFrom”.

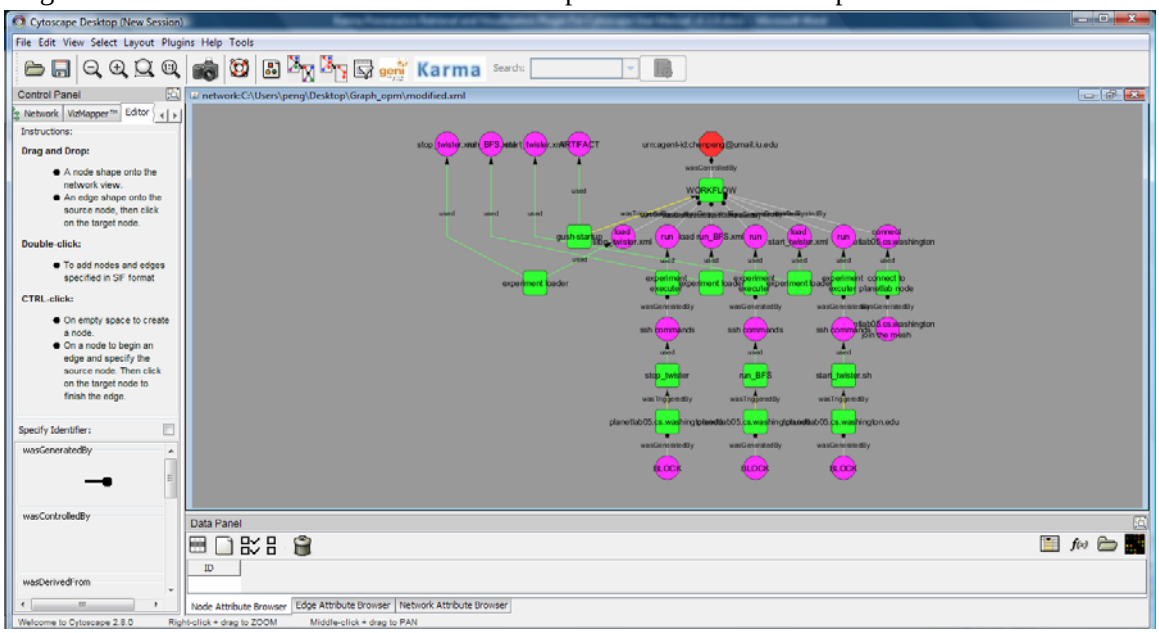

Figure 6.8.3.1 A provenance graph before applying “Compress Process”

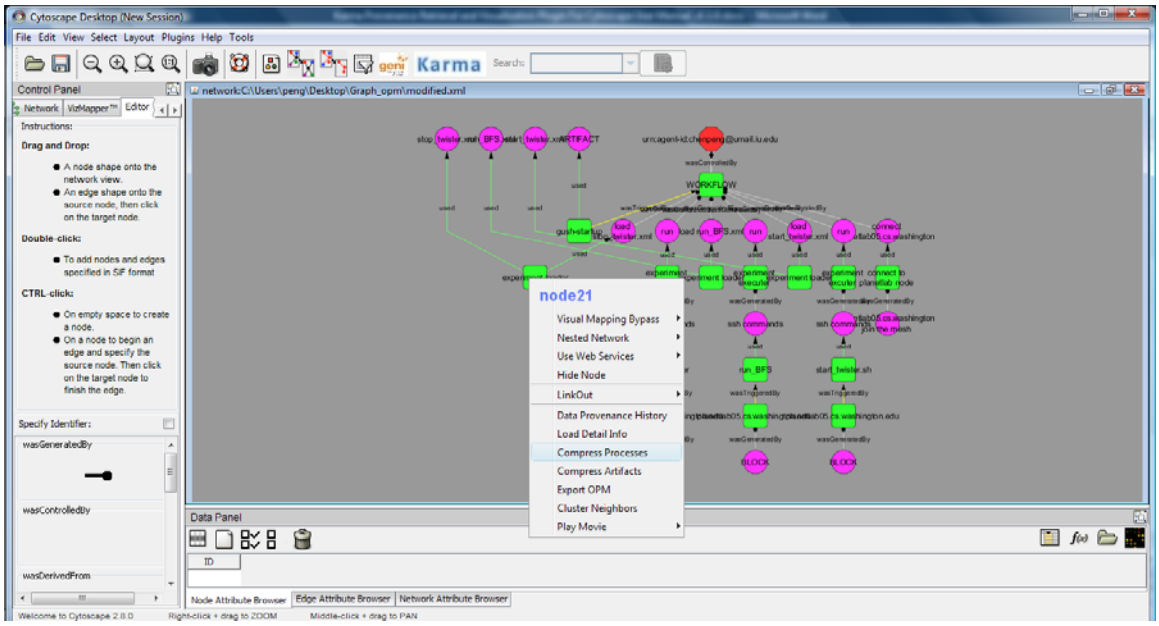

Figure 6.8.3.2 Applying “Compress Process”

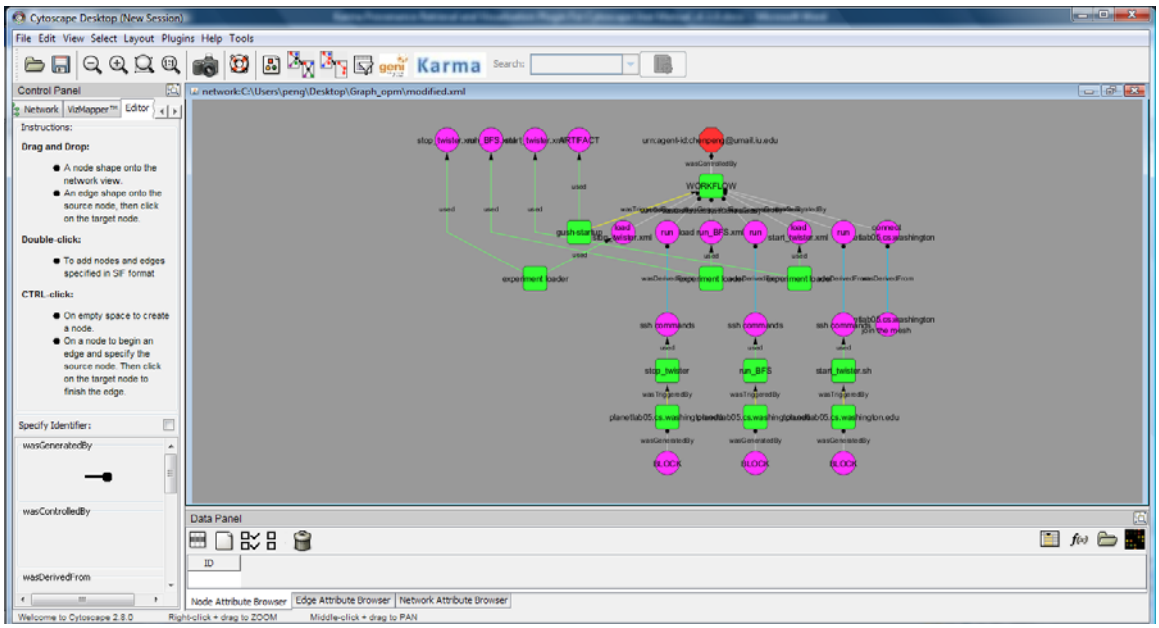

Figure 6.8.3.3 The provenance graph after applying “Compress Process”

Clicking on “Compress Artifact” will eliminate all of the “artifact” nodes that link two “Process” nodes where the outgoing edge represents the relationship “wasGeneratedBy” and the incoming edge represents the relationship “used”. The artifact will be replaced by a new edge between these two “Process” nodes that represents the relationship “wasTriggeredBy”.

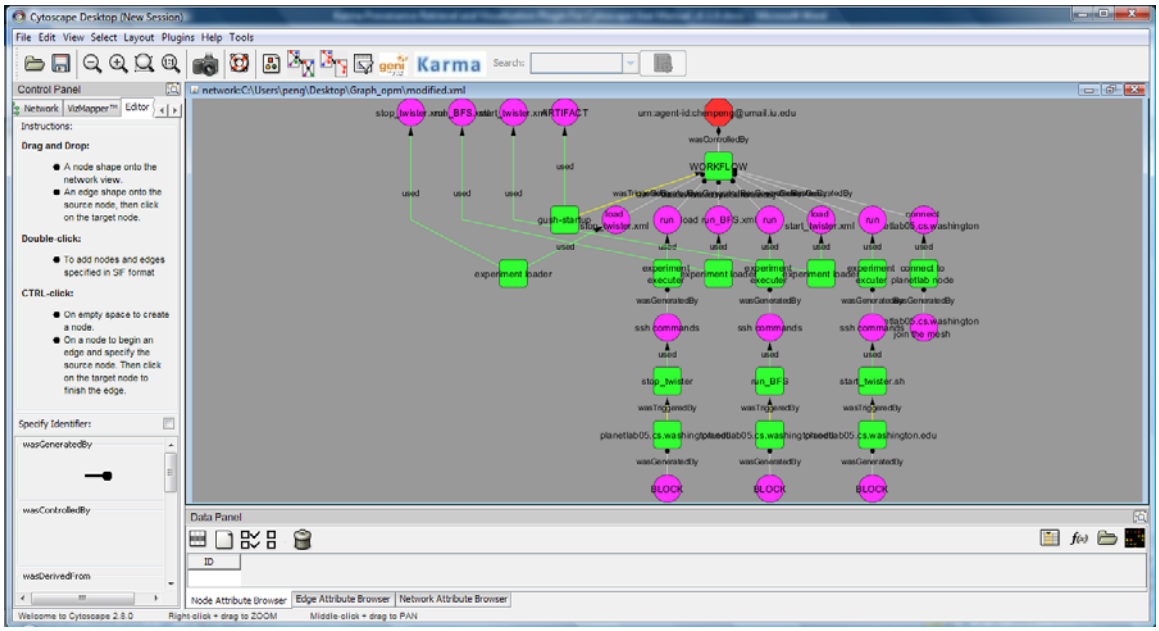

Figure 6.8.3.4 A provenance graph before applying “Compress Artifact”

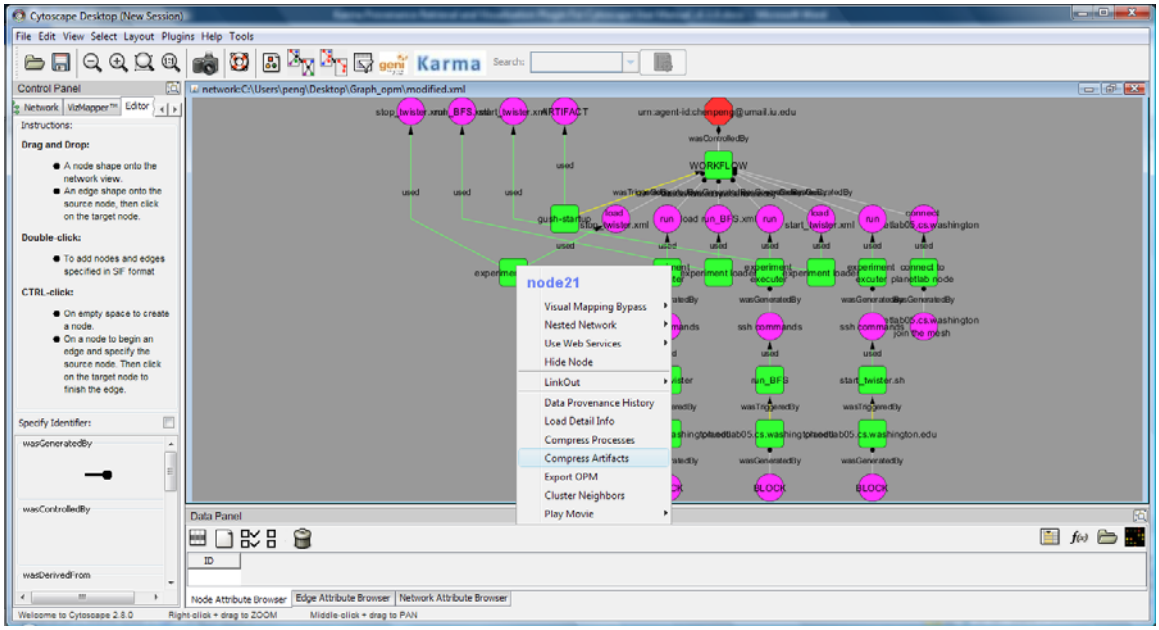

Figure 6.8.3.5 Applying “Compress Artifact”

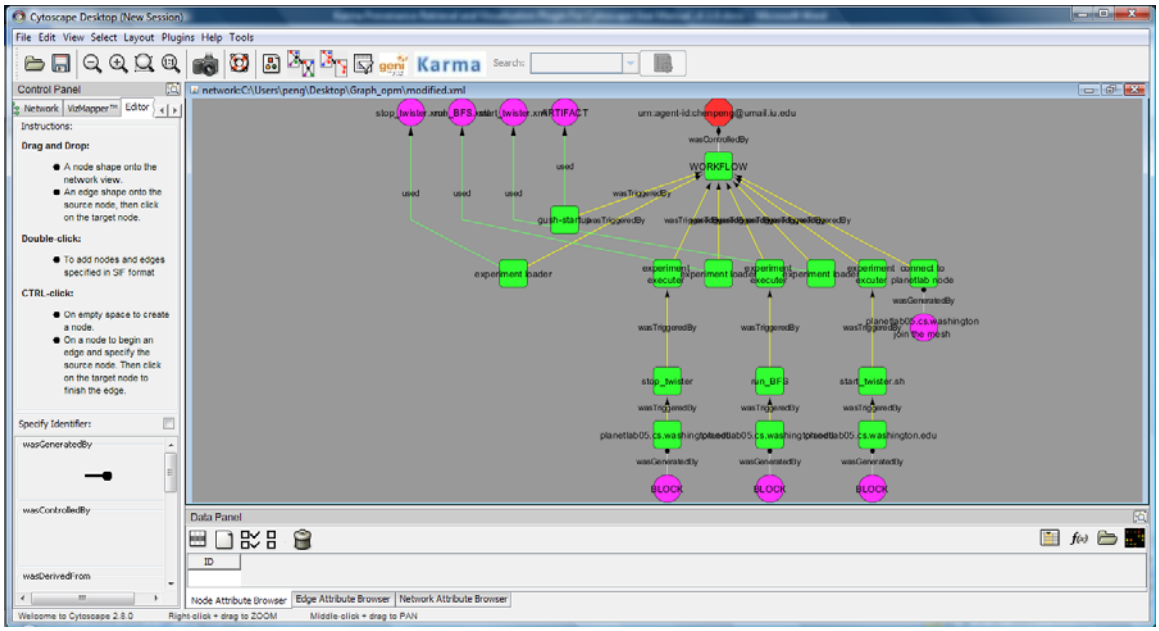

Figure 6.8.3.6 The provenance graph after applying “Compress Artifact”